PDnet™

PAD-800e Benutzerhandbuch

APEX

automation technologies GmbH

Einschränkung der Gewährleistung:

Die Angaben und Hinweise in diesem Handbuch sind sorgfältig geprüft und entsprechen den verfügbaren Daten zur Zeit der Drucklegung. Es wird keine Garantie für die Richtigkeit des Inhaltes übernommen.

Die im Handbuch verwendeten Software- und Hardwarebezeichnungen sind zum Teil eingetragene Warenzeichen und unterliegen als solche den gesetzlichen Bestimmungen.

Herausgeber:

APEX automation technologies GmbH
Vossenkamp 4
38104 Braunschweig

Telefon 0531-3704-0
Telefax 0531-3704-299

http://www.apex.de

CE

Der PDnetIP-Controller erfüllt die Anforderungen der EU-Richtlinie 89/336/EWG „Elektromagnetische Verträglichkeit". Die Konformität des PDnetIP-Controllers mit der o.g. Richtlinie wird durch das CE-Zeichen bestätigt.

INHALT

1 Einleitung

Dieses Handbuch unterstützt Sie beim Einbau, der Konfiguration und der Inbetriebnahme des PDnetIP-Controllers PAD-800e. Es beschreibt die Installation folgender Produktvarianten:

- **PAD-800e** verfügbar als
  PAD-800e/T, PAD-800e/L

Die APEX automation technologies GmbH behält sich Änderungen und Weiterentwicklungen vor, die zu Abweichungen von den in diesem Handbuch angegebenen Daten führen können.

Die Hardware wird von der APEX automation technologies GmbH zusammen mit einer PDnetIP-CD geliefert welche die für den Betrieb des PDnetIP-Controllers notwendige Software und Dokumentation beinhaltet.

Bestandteil der PDnetIP-CD ab Version 2.41 ist das Dokument „PDnetIP Versionsübersicht". Das Dokument gibt einen Überblick über die einzelnen Versionen.

Benutzen Sie die Software und Dokumentation einer PDnetIP-CD's immer als Einheit.

Die aktuelle PDnetIP-CD kann unter der Adresse ftp2.apex.de abgerufen werden. Benutzen Sie für den Download den Benutzernamen **kunde** und das Passwort **apex**.

1.1 Produktübersicht

Der PDnetIP-Controller PAD-800e ist ein intelligenter PMB-Teilnehmer. Mit der ICOS-Version des PAD-800e können Modnet-Koppler (nur SEA1) ersetzt werden und LOGISTAT CP80-A800 bzw. Logidyn D Automatisierungsgeräte an das PDnetIP (Ethernet, TCP/IP) angeschlossen werden.

Übersicht der PAD-800e

Artikel Nummer	Name	Beschreibung
20102060	PAD-800e/T	PDnetIP-Controller für LOGISTAT CP80-A800, mit RJ 45-Buchse
20102061	PAD-800e/T	PDnetIP-Controller für LOGISTAT CP80-A800, mit RJ 45-Buchse (inkl. ICOS)
20102062	PAD-800e/L	PDnetIP-Controller für LOGISTAT CP80-A800, mit RJ 45-Buchse und MT-RJ-Buchse
20102063	PAD-800e/L	PDnetIP-Controller für LOGISTAT CP80-A800, mit RJ 45-Buchse und MT-RJ-Buchse (ink.. ICOS)

Der PDnetIP-Controller ist in verschiedenen Versionen erhältlich. Aus lizenzrechtlichen Gründen darf ICOS nicht in jeder Version zur Verfügung stehen. Ergänzend zu den in der oberen Tabelle dargestellten Versionen, gibt es weitere optional erhältliche Module. Die folgende Tabelle gibt eine Übersicht, welche Protokolle lizenzrechtlich wann zur Verfügung stehen. Ob die gewünschte Kombination von Protokollen gleichzeitig nutzbar sind, ist separat zu prüfen.

Protokolle	Anmerkung
ModBus/TCP	Steht in jedem PDnetIP-Controller zur Verfügung.
VDM (PDnetIP)	
ICOS	Steht nur in den ICOS-Versionen zur Verfügung. Artikel-Nummern: 20102041 und 20102043
RFC1006	Separat zu erwerbende Lizenz
TCP-Schnittstelle	Separat zu erwerbende Lizenz
Seab 1/F Master	Separat zu erwerbende Lizenz
AccessServer	Separat zu erwerbende Lizenz. Notwendig für die Funktionen Fernbedienen und Fernladen in der A800.

1 Einleitung
1.1 Produktübersicht

Wichtig! Die benötigten Lizenzen müssen bereits bei der Bestellung des PDnetIP-Controllers angegeben werden. Zur Freischaltung weiterer Lizenzen bereits ausgelieferter Baugruppen müssen diese eingesandt werden.

Der PDnetIP-Controller ist mit wechselbaren Leitungsinterface-Modulen (LIM) bestückt. Die Leitungsart einer seriellen Schnittstelle wir über ein separates LIM-Modul bestimmt.

Die Lieferung erfolgt mit bestückten LIM-232. Die LIM-Module können durch den Anwender auf dem PDnetIP-Controller gewechselt werden.

Übersicht der Leitungsinterfacemodule

Artikel Nummer	Name	Beschreibung
10213010	LIM-232	LIM-232 Leitungs-Interface-Modul für RS-232 ohne Potentialtrennung
10213011	LIM-422	Leitungs-Interface-Modul für RS-422 ohne Potentialtrennung
10213012	LIM-485	Leitungs-Interface-Modul für RS-485 ohne Potentialtrennung
10213016	LIM-20 mA	Leitungs-Interface-Modul für 20 mA Stromschleife mit Potentialtrennung

1.2 Bestimmungsgemäße Verwendung

Der PDnetIP-Controller PAD-800e ist ausschließlich dazu bestimmt:

- eine LOGISTAT CP80-A800 oder Logidyn D
  mit dem PDnetIP zu verbinden.

Wird der PAD-800e anders als in diesem Handbuch beschrieben verwendet, ist kein sicherer Betrieb gewährleistet.

Für Schäden, Mehraufwendungen, Probleme, Störungen und insbesondere Personen- und Sachschäden, die aus nicht bestimmungsgemäßer Verwendung entstehen, ist der Hersteller nicht verantwortlich.

1.3 Allgemeine Hinweise

Der Einbau des PDnetIP-Controllers sollte von einer Fachperson durchgeführt werden. Die APEX automation technologies GmbH trägt keine Verantwortung für Schäden, die durch einen unsachgemäßen Einbau entstehen und/oder entstanden sind.

Das vorliegende Benutzerhandbuch erläutert den Einbau und die Inbetriebnahme der PDnetIP-Controller. Informationen zu den Steuerungssystem bzw. Programmiersystemen, sind den entsprechenden Handbüchern zu entnehmen.

Der PDnetIP-Controller sollte bis zu seinem endgültigen Einbau in dem Antistatik-Schutzbeutel verbleiben.

1.4 Lieferumfang

Zum Lieferumfang der PAD-800e gehören:

1. PAD-800e in Antistatikverpackung.
2. Konfigurationsblatt
   (Seriennummer, MAC-Adresse, geladene Firmware)
3. CD-ROM PDnetIP.

1.5 Ausbau und Lagerung

Wird der PDnetIP-Controller aus der Steuerung ausgebaut, muss dieser in einer geeigneten Verpackung (Antistatikverpackung) gelagert werden. Die unverpackte Lagerung oder Verpackung in Aluminiumfolie ist ungeeignet.

1.6 Reparatur

Bitte beachten Sie, dass im Falle einer Reparatur, der PDnetIP-Controller im Rahmen der Überprüfung mit der aktuellen Firmware und anderen Setupdaten geladen wird.

Es ist daher wichtig, alle Unterlagen und Werkzeuge zu besitzen, um die alte Konfiguration wieder herstellen zu können.

2 Technische Daten

Zuordnung	
Zentralbaugruppen	Alle Zentralbaugruppen mit 8-Bit PMB-Schnittstelle: COP032, COP132, ALU812, ALU981
Anschluss	PMB-Schnittstelle (über DPM) HW-Rev. 1..3: 8 KByte DPM HW-Rev. 4..5: 4 oder 8 KByte DPM Die Adresslage des DPM ist im Adressraum des PMB wählbar.
Versorgung	
Betriebsspannung	5 V / 900 mA (vom PMB)
PDnetIP-Anschlüsse	
Anzahl	1
Übertragungsge- schwindigkeit	10/100 Mbit/Sekunde Ethernet 10Base-T, 100Base-TX, 100Base-FX (LWL)
Kabeltyp	Twisted-Pair mit RJ-45 Stecker Cat. 5 LWL mit MT-RJ Anschluss
Serielle Schnittstellen	
Anzahl	2
Leitungsphysik	Wählbar über Steckmodul: RS-232, RS-485, RS-422, 20 mA
Übertragungsge- schwindigkeit.	50 — 115.200 Bit/Sekunde (die nutzbaren Baudraten sind in den Protokollen verschieden)
Anschluss	9-polige SubD-Buchse
Prozessor	
Typ	Intel 386EX
Speicher	HW-Rev. 1, 2: 1MB SRAM und 1MB Flashmemory HW-Rev. 3, 4, 5: 2MB SRAM und 2MB Flashmemory
Mechanischer Aufbau	
Kartenformat	Doppeleuropakarte (234 mm x 160 mm)
Breite	4 TE
Gewicht	350 g
Umgebungsbedingungen	
Temperatur	0 – 60 °C (im Betrieb), -40 - +85 °C (Lagerung)
Luftfeuchtigkeit	10 % — 80 % (nicht kondensierend)

Es können gleichzeitig mehrere PAD-800e in einem Magazin betrieben werden.

2.1 Hardware-Revisionen

Der PDnetIP-Controller PAD-800e existiert in verschiedenen Hardware-Revisionen. Die folgende Tabelle gibt einen Überblick über die existierenden Hardware-Revisionen. Die vorliegende Hardware-Revision kann folgendermaßen erkannt werden.

- Auf dem Steckverbinder des PDnetIP-Controllers ist die Hardware-Revision aufgedruckt.
- Die Kennzeichnung der Leiterplatte ist für jede Hardware-Revision individuell verschieden.
- Mit dem Programm DLSE32 kann die Hardware-Revision abgefragt werden.

Hardware-Revisionen (HW-Rev)

HW-Rev	Platinen-nummer	Bemerkung
1	200012097	Diese Hardware-Revision ist nicht mehr im Einsatz und funktioniert ausschließlich mit den dafür gelieferten Firmware-Versionen. Ein Update auf die aktuelle Firmware ist nicht möglich.
2	200049097	-
3	200429097	Diese Hardware-Revision benötigt mindestens die Firmware-Version 4.08.26.26
4	200839097	Diese Hardware-Revision benötigt mindestens die Firmware 4.08.26.26 und für die Unterstützung der 4k-Einstellung des DPM mindestens die Firmware 4.10.51.01.
5	201249097	-

Wichtig: Soll eine bereits installierte Karte durch eine neue ausgetauscht werden, ist die Kompatibilität zur vorhandenen Installation unbedingt zu prüfen. Ein älterer Firmware-Stand ist generell nicht auf neueren Karten nutzbar. Hingegen kann im allgemeinen ein neuerer Software-Stand auf älteren Karten (minimal: Hardware-Revision 2) eingesetzt werden.

2.2 Schnittstellen

Nachfolgend finden Sie Informationen zur Ausprägung der Schnittstellen. Protokollspezifische Details sind in den entsprechenden Absätzen zu den Protokollen erläutert.

2.2.1 PMB-Bus

Der PAD-800e tauscht über die PMB-Schnittstelle Daten mit der ALU der Steuerung aus. Die Adresse, auf welcher die Karte ihren DPM einblendet, wird mit den DIP-Schaltern konfiguriert. Details zur Adresseinstellung befinden sich in Kapitel 8.2.

2.2.2 Ethernet

Der PAD-800e/L ist mit zwei Schnittstellen ausgestattet. Er besitzt sowohl einen RJ45-Anschluss, als auch ein MT-RJ-Anschluss. Zu einem Zeitpunkt kann nur ein Anschluss genutzt werden.

Wichtig! Wird der PDnetIP-Controller über den MT-RJ-Anschluss betrieben, so muss auf der Gegenstelle als Betriebsart Halb-Duplex konfiguriert werden. Wird der PAD-800e über den RJ45-Anschluss genutzt, sollte dieser im Halbduplex-Mode betrieben werden. Im Vollduplex-Mode gibt es Kollisionszustände, welche nicht erkannt werden können, was wiederum zu Telegramm-Wiederholungen führt. Die Ursache liegt darin, dass auch in einem geswitchten Ethernet Kollisionen entstehen können.

Die Mac-Adresse (Ethernet-Adresse) des PDnetIP-Controllers ist in der Hardware gespeichert und unveränderlich. Alle gelieferten PDnetIP-Controller nutzen eine mit `00:05:DA` beginnende Mac-Adresse.

Die folgende Tabelle zeigt die Belegung der RJ-45 Anschlussbuchse, die der IEEE802.3 Twisted Pair-Schnittstelle entspricht.

Pin	Signal
1	Transmit +
2	Transmit -
3	Receive +
6	Receive -

2.2.3 Serielle Schnittstellen

Der PAD-800e ist mit 2 seriellen Schnittstellen ausgestattet. Die Schnittstellen sind als 9-polige SubD-Buchsen auf der Frontseite ausgeführt. Die obere SubD-Buchse ist die erste serielle Schnittstelle und entsprechend die untere die zweite serielle Schnittstelle.

Die Leitungsinterfacemodule (LIM-Module) passen die seriellen Schnittstellen des PDnetIP-Controllers an die Leitungsphysik an. Die Standardausstattung erfolgt mit bestückten LIM-232.

Die LIM-Module können durch den Anwender auf dem PDnetIP-Controller gewechselt werden.

Die folgende Tabelle zeigt die PIN-Belegung der seriellen Schnittstellen in Abhängigkeit der eingesetzten LIM-Module.

PIN Belegung der seriellen Schnittstellen

Pin	LIM-232 V.24 Signal	LIM 485 RS 485	LIM-422 RS 422	LIM-20 mA Stromschleife
1	DCD	—	—	Tx-
2	RxD	TxD +	TxD +	Rx +
3	TxD	RxD +	RxD +	Tx +
4	DTR	+5V	+5V	—
5	GND	GND	GND	GND
6	—	—	—	Rx -
7	RTS	RxD-	RxD-	RTS (TTL)
8	CTS	TxD-	TxD-	CTS (TTL)
9	—	—	—	—

— nicht belegt

Die folgende Tabelle zeigt wie ein Programmierkabel zu konfektionieren ist, um den PDnetIP-Controller über die serielle Schnittstelle mit dem Programm DLSE32 laden zu können. Alternativ kann das Standard-Programmierkabel YDL52 genutzt werden.

Die nachfolgende Tabelle dokumentiert die minimale notwendige Verkabelung für ein Programmierkabel.

PDnetIP-Controller	PC
2	3
3	2
5	5

3 Anzeige- und Bedienelemente

3.1 Aufteilung der Front

Je nach Hardware-Revision ist die Front inklusive der Leuchtdioden des PAD-800e verschieden gestaltet. Nachfolgend finden sich, nach Hardware-Revision getrennt, die Informationen welche Leuchtdioden vorhanden sind. Dabei wird jeder Leuchtdiode ein Kennung zugeordnet. Die Beschreibung der einzelnen Leuchtdioden erfolgt ab Kapitel 3.2.

Die nachfolgende beschriebenen Leuchtdioden sind von oben nach unten sowie von links nach rechts aufgeführt.

3 Anzeige- und Bedienelemente

3.1 Aufteilung der Front

3.1.1 Hardware-Revision 1, 2

Zur Diagnose ohne spezielle Hilfsmittel besitzt der PDnetIP-Controller 16 Leuchtdioden (LED) zur Darstellung des aktuellen Betriebszustandes.

Kennung	Beschriftung	Farbe	Bemerkung
E1	100	Grün	
E2	FDX	Grün	
E3	ACT	Grün	
E4	LNK	Grün	
S1	PAD-Sts	Gelb	
S2	Hardw.	Gelb	
S3	Setup	Gelb	
S4	Rem.PG	Gelb	
S5	Bus-Sts	Gelb	
S6	Ser.1Sts	Gelb	
S7	Ser.2Sts	Gelb	
S8	DPM-Sts.	Gelb	
C1	RD	Grün	Oberhalb von Ser. 1
C2	TD	Rot	
C3	RD	Grün	Oberhalb von Ser. 2
C4	TD	Rot	

3 Anzeige- und Bedienelemente
3.1 Aufteilung der Front

3.1.2 Hardware-Revision 3, 4, 5

Zur Diagnose ohne spezielle Hilfsmittel besitzt der PDnetIP-Controller 17 Leuchtdioden (LED) zur Darstellung des aktuellen Betriebszustandes.

Kennung	Beschriftung	Farbe	Bemerkung
E1	100	Grün	
E2	FDX	Grün	
E3	RX oder RD	Grün	
E4	LNK	Grün	
S1	PAD-Sts	Gelb	
S2	Hardw.	Gelb	
S3	Setup	Gelb	
S4	Rem.PG	Gelb	
S5	Bus-Sts	Gelb	
S6	Ser.1Sts	Gelb	
S7	Ser.2Sts	Gelb	
S8		Gelb	Unterhalb von LED S7
S8B		Blau	Rechts unterhalb von LED S7
C1	RD	Grün	Oberhalb von Ser. 1
C2	TD	Rot	
C3	RD	Grün	Oberhalb von Ser. 2
C4	TD	Rot	

3.2 Bedeutung der Leuchtdioden

3.2.1 Ethernet Status, LED E1..E4

Die vier grünen Leuchtdioden E1 bis E4 dienen der Diagnose des Hardware-Status der Ethernet-Schnittstelle. Die Steuerung der Leuchtdioden erfolgt direkt von der Hardware und ist von der Firmware unabhängig. Erfolgt die Verbindung über die RJ45-Schnittstelle, zeigen die Leuchtdioden den aktuellen Status an. Je nach Verbindungsart wechselt die Anzeige.

Erfolgt die Verbindung über den MT-RJ-Anschluss (PAD-800e/L), zeigen die Leuchtdioden die eingestellte Betriebsart des PDnetIP-Controllers an. Dies geschieht unabhängig von der Betriebsart der Gegenstelle.

ID	Inhalt	Zustand	Bedeutung
E1	100 Mbit	an	100-Mbit Verbindung
		aus	10-Mbit Verbindung
E2	Full-Duplex	an	Voll-Duplex Verbindung
		aus	Halb-Duplex Verbindung
E3	Empfangs-daten	flackert	PDnetIP-Controller empfängt Daten. Sendedaten werden über die LED nicht angezeigt.
		aus	Keine Empfangsdaten vorhanden.
E4	LINK	an	Verbindung vorhanden
		aus	Keine Verbindung vorhanden

3.2.2 LED S1..S8/S8B

Die gelben Leuchtdioden S1 bis S8 bzw. S8B (Hardware-Revision: 3, 4, 5) dienen der Diagnose des Firmware- oder Lader-Status. Ist der Taster nicht gedrückt und die Leuchtdiode S4 aus läuft eine Firmware und es gilt die Beschreibung aus Kapitel 3.2.2.1, ist die Leichtdiode S4 an oder blinkt sie ist der PDnetIP-Controller im Lader und es gilt die Beschreibung aus Kapitel 3.2.2.2.

3.2.2.1 Firmware Status

Die Bedeutung der einzelnen Leuchtdioden ist teilweise abhängig von der in den PDnetIP-Controller geladenen Projektierung.

ID	Inhalt	Zustand	Bedeutung
S1	PAD Status	an	Normalzustand
		aus	Falsche Firmwareversion
		blinkt	Ein Firmwaremodul ohne eigene Diagnose LED meldet einen Fehler.
S2	Hardware	aus	Normalzustand, kein Fehler.
		blinkt	Die Firmware hat einen Hardwarefehler erkannt.
S3	Setup	an	Der PDnetIP-Controller besitzt gültige Setupdaten.
		blinkt	Die Setupdaten sind fehlerhaft oder nicht vorhanden.
S4	Remote PG	aus	Normaler Betriebszustand.
		an	Der Lader läuft und es gilt die Tabelle in Kapitel 3.2.2.2.
		blinkt	
S5	Bus Status	an	Ethernet Verbindung vorhanden.
		blinkt	Kein Link am Ethernet-Port vorhanden.
S6 S7	Status serielle Schnittstelle 1 / 2	aus	Es wurde kein Treiber für die serielle Schnittstelle geladen.
		an	Es wurde ein Treiber für die serielle Schnittstelle geladen.
		blinkt	Fehler in der Kommunikation der seriellen Schnittstelle.
S8	DPM Status	an	Die Kommunikation mit der SPS wurde aktiviert.
		aus	Es wurde kein Endgerätetreiber zur Kommunikation mit der SPS aktiviert.
		blinkt	Die Verbindung zwischen PDnetIP-Controller und SPS ist oder war gestört.
S8B	ICOS	an	Das ICOS-Interface wurde erfolgreich initialisiert.
		aus	Es wurde kein ICOS-Interface initialisiert.

Ursachen für Fehlermeldungen

ID	Mögliche Ursache
S1 PAD Status	Eine Struktur soll mit dem SFB693 übertragen werden, wurde aber zuvor nicht angemeldet. Damit kennt der PAD nicht die Adresslage der Struktur und kann diese nicht übertragen. Maßnahme: SPS-Programm ändern
	Der Endgerätetreiber soll eine Struktur übertragen, aber unterstützt keine Strukturen. Maßnahme: Projektierung korrigieren
	Die Konfigurationsinformationen sind zerstört (wenn zugleich S2 blinkt) Maßnahme: Reparatur notwendig
S2 Hardware	Der Endgerätetreiber hat Probleme mit dem Interruptbetrieb (Hardware defekt). Wenn das Problem vorhanden ist blinkt die LED ca. 10 Sekunden nachdem das SPS-Programm gestartet wurde. Maßnahme: Reparatur notwendig
	Die Konfigurationsinformationen sind zerstört (wenn zugleich S1 blinkt) Maßnahme: Reparatur notwendig
	Die Konfiguration der Hardware sollte aktualisiert werden. Dieser Fehler tritt in seltenen Fällen auf wenn eine Aktualisierung weitere Möglichkeiten eröffnet.
S3 Setup	ICOS konfiguriert aber nicht linzensiert Maßnahme: Projektierung korrigieren oder Lizenz erwerben
	RFC1006 konfiguriert aber nicht linzensiert Maßnahme: Projektierung korrigieren oder Lizenz erwerben
	TCP-Interface konfiguriert aber nicht linzensiert Maßnahme: Projektierung korrigieren oder Lizenz erwerben
	Datenzelle zu groß Maßnahme: Projektierung korrigieren
	Setupdaten fehlen oder sind defekt (wenn S4..S8 den Zustand „aus“ haben) Maßnahme: Setupdaten programmieren
	Setupdaten enthalten nicht unterstützten Treiber Maßnahme: NetPro/Firmware-Kombination prüfen
	Ein Modul der Setupdaten liegt in einer anderen Version vor als es von der Firmware unterstützt wird Maßnahme: NetPro/Firmware-Kombination prüfen
S5 BUS Status	kein Link am Ethernet-Port vorhanden Maßnahme: Ethernet-Konfiguration prüfen
S8 DPM Status	Ein ICOS Sendetelegramm wurde nicht rechtzeitig von der Partnerstation quittiert
	Ein ICOS Sendetelegramm kann nicht zugestellt werden, weil keine Verbindung zur Partnerstation aufgebaut werden kann.
	Der Datenaustausch zwischen Endgerät und PAD-800e ist gestört (SPS steht bzw. Baustein wird nicht aufgerufen). Maßnahme: SPS Programm ändern, SPS Programm starten, ggf. ICOS Initialisierung durchführen

Wichtig! Die Ursache für eine Fehlermeldung kann leichter festgestellt werden wenn ein SYSLOG-Server vorhanden ist (siehe Kapitel 4.3.6).

3.2.2.2 Lader Status

Der Lader benötigt nur einen Teil der Leuchtdioden S1..S8B. Dadurch bedingt ist der Normalzustand einiger Leuchtdioden „AUS“.

<table>
<tr><th>ID</th><th>Inhalt</th><th>Zustand</th><th>Bedeutung</th></tr>
<tr><td>S1</td><td>PAD Status</td><td>aus</td><td>Normalzustand</td></tr>
<tr><td rowspan="2">S2</td><td rowspan="2">Hardware</td><td>aus</td><td>Normalzustand</td></tr>
<tr><td>blinkt</td><td>Der Lader hat einen Fehler erkannt.</td></tr>
<tr><td>S3</td><td>Setup</td><td>aus</td><td>Normalzustand</td></tr>
<tr><td rowspan="3">S4</td><td rowspan="3">Remote PG</td><td>aus</td><td>Die Firmware läuft und es gilt Tabelle aus Kapitel 3.2.2.1.</td></tr>
<tr><td>an</td><td>Der Lader ist aktiv und arbeitet ausschließlich mit der seriellen Schnittstelle über welche er die Programmierdaten empfangen hat. Die andere serielle Schnittstelle wurde vom Lader deaktiviert.</td></tr>
<tr><td>blinkt</td><td>Der Lader ist aktiv und hat noch keine Programmierdaten empfangen.</td></tr>
<tr><td>S5</td><td>Bus Status</td><td>aus</td><td>Normalzustand</td></tr>
<tr><td rowspan="2">S6
S7</td><td rowspan="2">Status serielle Schnittstelle 1 / 2</td><td>aus</td><td>Die serielle Schnittstelle wurde vom Lader deaktiviert. Um die Schnittstelle wieder nutzen zu können ist ein Neustart des PDnetIP-Controllers notwendig.</td></tr>
<tr><td>an</td><td>Wenn zugleich die Leuchtdiode S4 (Remote PG) blinkt ist der Lader bereit über die serielle Schnittstelle Daten zu empfangen, ist die Leuchtdiode „AN“ so empfängt der Lader ausschließlich über diese serielle Schnittstelle Daten.</td></tr>
<tr><td>S8</td><td>DPM Status</td><td>aus</td><td>Normalzustand</td></tr>
<tr><td>S8B</td><td>ICOS</td><td>aus</td><td>Normalzustand</td></tr>
</table>

Ursachen für Fehlermeldungen

<table>
<tr><th>ID</th><th>Mögliche Ursache</th></tr>
<tr><td rowspan="4">S2
Hardware</td><td>Nach dem Programmstart des Laders oder während der seriellen Kommunikation hat der Lader Fehler im Zugriff auf den Flashmemory festgestellt.
Maßnahme: Reparatur notwendig.</td></tr>
<tr><td>Der Lader hat eine nicht vorgesehene Meldung erhalten.
Maßnahme: PDnetIP-Controller nochmals starten (Power-Off/On). Steht der Fehler weiterhin an ist eine Reparatur notwendig.</td></tr>
<tr><td>Der Lader kann die Hardwarekonfiguration des PDnetIP-Controllers nicht erkennen. Die Konfigurationsdaten sind nicht lesbar oder zerstört.
Maßnahme: Reparatur notwendig</td></tr>
<tr><td>Der Lader unterstützt die Hardware-Konfiguration des PDnetIP-Controllers nicht. Eine ungeeignete (alte) Version des Laders wurde in einem (neueren) PDnetIP-Controller geladen.
Maßnahme: Reparatur notwendig</td></tr>
</table>

3.2.3 Serielle Schnittstellen, LED C1..C4

Die Leuchtdioden C1 bis C4 dienen der Diagnose der seriellen Schnittstellen. Die Steuerung der Leuchtdioden erfolgt direkt von der Hardware und ist von der Firmware unabhängig. Der Datenempfang der seriellen Schnittstellen kann durch ein Aufleuchten der entsprechenden LED (C2 bzw. C4) geprüft werden.

Nr.	Inhalt	Zustand	Bedeutung
C1	Sendedaten Seriell 1	an	Es werden Daten gesendet.
		aus	Es werden keine Daten gesendet
C2	Empfangsdaten Seriell 1	an	Es werden Daten empfangen.
		aus	Es werden keine Daten empfangen.
C3	Sendedaten Seriell 2	an	Es werden Daten gesendet
		aus	Es werden keine Daten gesendet
C4	Empfangsdaten Seriell 2	an	Es werden Daten empfangen.
		aus	Es werden keine Daten empfangen.

3.3 Taster

Der Taster hat verschiedene Funktionen. Je nach Tastzeit und aktiver Betriebsart werden die unterschiedlichen Funktionen aktiviert:

- Fehler quittieren → siehe Kapitel 3.3.1
- Lader aktivieren → siehe Kapitel 3.3.2.1
- Lader verlassen → siehe Kapitel 3.3.2.2
- Neustart veranlassen → siehe Kapitel 3.3.3

3.3.1 Fehler quittieren

Durch das kurze Drücken des Tasters (maximal eine Sekunde) werden die internen Fehlermerker der Firmware gelöscht. Damit verbunden wechseln die Leuchtdioden von S1 bis S8 die blinken in den Zustand Dauerlicht. Steht die Fehlerursache weiterhin an, so beginnt die der Fehlerursache zugeordnete Leuchtdiode nach einer Zeit erneut zu blinken.

3.3.2 Lader

Je nachdem ob der Lader aktiviert oder verlassen werden soll muss man zuvor erkennen, ob der Lader oder eine Firmware läuft. Die Unterscheidung ist am einfachsten an der Leuchtdiode S4 (Remote-PG) möglich. Ist der Taster nicht gedrückt und die Leuchtdiode S4 aus läuft eine Firmware, ist die Leuchtdiode an oder blinkt, ist der PDnetIP-Controller im Lader.

3.3.2.1 Aktivieren

Die folgende Tabelle zeigt, wie der Lader aktiviert wird, wenn eine Firmware läuft. In jeder Spalte ist das Abbild der gelben Leuchtdioden S1 (oben) bis S8 (unten) markiert.

	1	2	3	4	5	6
S1	●	○	●	●	●	○
S2	○	○	○	●	●	○
S3	●	○	○	●	●	○
S4	○	○	○	●	○	✱
S5	●	○	○	●	○	○
S6	○	○	○	●	○	●
S7	○	○	○	●	○	●
S8	●	○	○	●	○	○

Spalte	Bedeutung
1	Es läuft eine Firmware und es stehen keine Fehler an.
2	Der Taster wurde gedrückt und es gehen alle Leuchtdioden aus.
3	Im Sekundentakt gehen weitere Leuchtdioden an.
4	Nach acht Sekunden sind alle Leuchtdioden an, damit wurde die Firmware informiert, dass ein Neustart durchgeführt werden soll.
5	Der PDnetIP-Controller führt einen Neustart durch und es gehen wiederum alle Leuchtdioden der Reihe nach an.
6	Anschließend befindet sich der PDnetIP-Controller im Lader, was durch das Blinken der Leuchtdiode S4 (Remote-PG) erkennbar ist.

3.3.2.2 Verlassen

Die folgende Tabelle zeigt, wie ein aktivierter Lader wieder verlassen werden kann. In jeder Spalte ist das Abbild der gelben Leuchtdioden S1 (oben) bis S8 (unten) markiert.

	1	2	3	4	5	6
S1	○	●	○	○	○	●
S2	○	○	●	○	○	○
S3	○	○	○	○	○	●
S4	✱	○	○	○	○	○
S5	○	○	○	○	○	●
S6	●	○	○	○	○	○
S7	●	○	○	○	○	○
S8	○	○	○	●	○	●

Spalte	Bedeutung
1	Es läuft ein Lader.
2	Der Taster wurde gedrückt und es gehen die Leuchtdioden S1 bis S8 aus.
3	Etwas schneller als im Sekundentakt wandert die Leuchtdiode nach „unten“.
4	Alle gelben Leuchtdioden S1 bis S8 wurden durchlaufen.
5	Wenn die letzte gelbe Leuchtdiode S8 erlischt, führt der PDnetIP-Controller einen Neustart durch. In diesem Moment muss der Taster losgelassen werden, damit die Firmware starten kann.
6	Anschließend läuft im PDnetIP-Controller die Firmware.

3.3.3 Neustart

Soll der PDnetIP-Controller einen Neustart durchführen, ist der Taster solange zu drücken, bis alle Leuchtdioden S1..S8 den Zustand AUS haben. Anschließend muss der Taster sofort losgelassen werden. Der PDnetIP-Controller führt einen Neustart durch.

4 Firmware

Der Leistungsumfang der PDnetIP-Controller wird im wesentlichen von der verwendeten Firmware bestimmt. Alle hier gemachten Angaben beziehen sich auf den Firmwarestand 4.10.51.01, welcher zum Zeitpunkt der Redaktion dieses Handbuches aktuell ist.

Genaue Informationen zum Leistungsumfang sind ausschließlich für einzelne Firmwarestände möglich und den jeweiligen Versionsinformationen zu entnehmen. Andere Firmware-Versionen können abweichende Leistungsmerkmale aufweisen.

Achtung! Um die Dienste ICOS, RFC1006 oder die freie TCP-Schnittstelle nutzen zu können, müssen diese auf der Karte freigeschaltet sein. Der nachfolgende Text geht immer von der Verfügbarkeit der entsprechenden Dienste aus.

Achtung! Die im Handbuch erwähnte Kommunikation von Strukturen steht derzeit im PAD-800e nicht zur Verfügung.

4.1 Allgemeine Leistungsmerkmale

Protokoll	Limit
ARP	64 ARP-Einträge
DNS	2 Server
ICOS/TCP	32 Verbindungen
ModBus/TCP	32 Verbindungen
RFC1006	50 Verbindungen
SYSLOG	1 Server
TCP-Schnittstelle	50 Verbindungen
VDM	50 Partnerstationen

Folgende Limits gelten für alle Protokolle in Summe

- 128 TCP-Verbindungen
- 5 UDP-Verbindungen

Eine Reserve von 10% sollte berücksichtigt werden.

4.2 TCP/UDP Ports

Der PDnetIP-Controller nutzt die in der Tabelle aufgeführten Port-Nummern. Diese Informationen sind für den Netzwerk-Administrator von Bedeutung, wenn aktive Netzwerkkomponenten wie Router oder Firewalls genutzt werden.

Protokoll	Port-Nummer
DNS	TCP 53
BOOTP/DHCP	UDP 68
ISO TP0 over TCP (RFC1006)	TCP 102
ModBus/TCP	TCP 502
SYSLOG	UDP 514
PDnetIP (VDM)	UDP/TCP 2843
ICOS/TCP	UDP/TCP 9876
TCP	Abhängig von der Projektierung
Access-Server	Abhängig von der Projektierung

4.3 Netzwerk-Protokolle

Der PDnetIP-Controller unterstützt auf dem Netzwerk-Interface die folgenden Protokolle:

- ICOS/TCP,
- MODBUS/TCP,
- RFC1006,
- Freies TCP-Protokoll,
- VDM.

Wichtig! Alle Protokolle außer ICOS/TCP benötigen den Baustein PDNIP, um auf den Signalspeicher der Steuerung zugreifen zu können.

Je nach Anwendungsfall ist zu entscheiden, mit welchem Protokoll die bestmögliche Lösung implementiert werden kann. Die nachfolgende Tabelle gibt einen Überblick über Vorteile und Nachteile der einzelnen Protokolle.

Protokoll	Vorteil/Nachteil
ICOS/TCP	Ermöglicht die effiziente Ablösung vorhandener ICOS-Kommunikationssysteme. Durch Integration des PDM-Servers von der APEX automation technologies GmbH können Systeme mit OPC-Schnittstelle in die ICOS-Welt integriert werden. Der Datenaustausch wird wie bisher über Sende- und Empfangsbits abgewickelt.
MODBUS/TCP	MODBUS/TCP stellt ein einfach zu handhabendes Protokoll mit hohem Verbreitungsgrad dar. Durch die Verfügbarkeit von Gateways ist die Integration von Stationen mit ModBus/Plus oder ModBus/Seriell möglich. Werden Leitsysteme mit MODBUS/TCP angebunden, ist die Aktualisierungsrate im Vorfeld zu betrachten, da die Datenübertragung im allgemeinen pollend erfolgt. Der Datenaustausch via MODBUS/Master wird von Anwenderprogramm der SPS aus gesteuert.
RFC1006	Mit RFC1006 können leicht größere Datenmengen ausgetauscht werden. Die Handhabung des Protokolls ist aufwendiger als bei anderen Protokollen. Der Datenaustausch erfolgt dabei über vorprojektierte Sende-, Empfangspuffer sowie Sende- und Empfangsbits.
Freies TCP-Protokoll	Mit dem freien TCP-Protokoll können individuelle Anbindungen an Fremdsysteme implementiert werden. Da jedoch kein eigentliches Protokoll standardisiert ist und die Daten 1:1 ausgetauscht werden, gibt es im allgemeinen höhere Anforderungen an die SPS-Implementierung.
VDM	Der VDM ermöglicht einen effizienten und ereignisorientierten Datenaustausch mit anderen PDnetIP-Stationen. Dabei wird die gesamte Kommunikation mit NetPro zentral projektiert, im Anwenderprogramm werden keine Kommunikationsmodule benötigt. Die Integration von Systemen mit OPC-Schnittstelle ist mit dem PDM-Server möglich.

4.3.1 ICOS/TCP

Der PDnetIP-Controller ist in der Lage, ICOS-Telegramme über TCP/IP zu transportieren und so ältere Netzwerke, wie Modnet-2NP, 2ND, ICOS via H1 u.ä., zu substituieren. In diesem Fall läuft die Funktionalität des alten Netzes als Dienst auf dem TCP/IP-Netz mit. Zu diesem Zweck stellt der PDnetIP-Controller der SPS ein ihr „bekanntes“ Interface bereit, über welches die Steuerung den Datenaustausch abwickeln kann. Dies ermöglicht einen nahtlosen Übergang zu neuen Lösungen. Dabei erweitert der PDnetIP-Controller das herkömmliche Interface um weitere Leistungsmerkmale.

Als Interface kann eines der folgenden genutzt werden: KP92 oder KOS-881. Dabei hat das KP92-Interface die größte Flexibilität und Leistungsfähigkeit. Je nach verwendeter Hardware/Firmware kann auch ein KOSKO-Interface genutzt werden.

Es werden alle ICOS-Telegrammtypen inkl. B8EFELD ohne Einschränkungen hinsichtlich der Telegrammlänge unterstützt.

Der PDnetIP-Controller kann zu einem Zeitpunkt nur eine limitierte Anzahl von ICOS-Verbindungen zu verschiedenen Partnerstationen aufrecht halten. Sind über eine IP-Adresse verschiedene A-Bytes erreichbar, zählt dies als eine Verbindung. Die maximale Anzahl Verbindungen ist im Kapitel 4.1 definiert.

Wichtig: Es müssen alle A-Bytes projektiert werden, mit welchen der PDnetIP-Controller kommunizieren soll. Dies trifft auch auf PC-Systeme zu, welche als externe Station definiert werden. Nur so kann der PDnetIP-Controller ein A-Byte in eine IP-Adresse bzw. einen DNS-Namen auflösen.

Die PDnetIP-Controller unterstützen die Verarbeitung folgender ICOS Prozeduren.

<table>
<tr><th>Prozedur</th><th>A-Byte</th><th>Broadcast-Adresse</th></tr>
<tr><td>BUS_1_MA</td><td>240</td><td rowspan="3">127</td></tr>
<tr><td>BUS_1_SL</td><td>1</td></tr>
<tr><td>BUS_2_TA</td><td rowspan="3">wie projektiert</td></tr>
<tr><td>BUS_2_TB</td><td rowspan="2">255</td></tr>
<tr><td>BUS_2_FM</td></tr>
<tr><td>STERN_1_MA</td><td>240</td><td rowspan="2">127</td></tr>
<tr><td>STERN_1_SL</td><td>1</td></tr>
</table>

Der PDnetIP-Controller unterstützt BUS- und STERN-Prozeduren, welche ursprünglich nur zur seriellen Kommunikation vorgesehen waren. Bei diesen Prozeduren gelten die in der Tabelle dargestellten Sonderregeln zur Bestimmung des A-Byte. Damit ist es dem PDnetIP-Controller möglich, die Prozeduren ohne Anpassungen auf der SPS-Seite zu unterstützen.

Wichtig! Alle BroadCast-Telegramme werden an die BroadCast-Adresse des Subnetzes gesendet. Damit können die Stationen, welche Mitglied des Subnetzes sind, BroadCast-Telegramme empfangen.

Wichtig! Wird die ICOS-Kommunikation von der SPS initialisiert, (Einschaltmoment oder via IK) übergibt die SPS das A-Byte an den PDnetIP-Controller. Das A-Byte wird vom PDnetIP-Controller mit dem projektierten A-Byte verglichen und nur bei einer Übereinstimmung wird die Initialisierung quittiert.

Wichtig! Erkennt der PDnetIP-Controller eine Verbindungsstörung zu einem PC bzw. einer externen Station wird diese Station gesperrt. Um die Verbindung zu „reaktivieren“ muss der PC bzw. die externe Station selbstständig die Verbindung neu aufbauen. Ist das Verhalten nicht gewünscht, muss die Station im NetPro als PDnetIP-Controller projektiert werden.

4.3.1.1 Fernbedienen und Fernladen

Die Funktionen Fernbedienen und Fernladen realisiert über eine serielle Verbindung des PAD-800e mit der Zentralbaugruppe der A800 einen Zugriff auf die Programmierfunktionen der Steuerung. Dafür wird der Treiber Access-Server auf der gewünschten seriellen Schnittstelle des PAD-800e projektiert.

Access-Server, serielle Schnittstelle 1
Allgemein
Profil A800 Programmiergerät
Port (TCP/UDP) 8001
OK Abbrechen Std-Werte

Für die Nutzung des Access-Server ist eine Freischaltung zu erwerben, das notwendige spezielle Verbindungskabel zwischen PAD-800e und ALU wird mitgeliefert.

4.3.2 ModBus/TCP

Der PDnetIP-Controller unterstützt ModBus/TCP Funktionen im Server- und Master-Betrieb.

Soll auf den Signalspeicher der A800 von außen zugegriffen werden (Leitsystem, Bediengeräte etc.), so wird dafür der Server-Betrieb benötigt (siehe Kapitel 4.3.2.2).

Soll die A800 selbst aktiv mit anderen ModBus/TCP-Stationen im Netzwerk kommunizieren (Master-Betrieb), werden gegenüber dem Server-Betrieb ergänzend die Auftragsblöcke benötigt (siehe Kapitel 4.3.2.3).

In manchen Systemen wird nicht vom ModBus Client sondern vom ModBus Slave gesprochen. An diesen Stellen ist im allgemeinen ein ModBus Client gemeint. Die Bezeichnung ModBus Slave wird oft aus historischen Gründen benutzt, existiert bei ModBus/TCP aber nicht mehr.

Der PDnetIP-Controller kann zu einem Zeitpunkt nur eine limitierte Anzahl von ModBus/TCP-Verbindungen aufrecht halten. Die maximale Anzahl Verbindungen ist im Kapitel 4.1 definiert.

Eine Station (IP-Adresse) kann gleichzeitig mehrere Verbindungen zu den PDnetIP-Controllern unterhalten und dadurch eine bessere Kommunikationsgeschwindigkeit erreichen.

4.3.2.1 ModBus Mapping

Viele Steuerungen kennen mehr Datentypen als die im ModBus/TCP-Standard definierten Coils und Register. Für eine flexible Anwendung der ModBus/TCP Kommunikation wurde das Modbus Mapping eingeführt. Ein Mapping ordnet einem ModBus/TCP Bereich einen Bereich des Signalspeichers der SPS zu.

Dabei definiert ein Mapping, wo die Daten aus ModBus-Sicht in der Steuerung liegen. Ein Mapping selbst führt zu keiner Kommunikation.

Zuordnung zwischen ModBus/TCP und Endgerät

ModBus/TCP
Datentyp: Holding Register
Adresse: 1
Größe: 4096

Endgerät
Datentyp: DPM-Offset
DPM-Offset: 0
Read-Only: Schreibschutz der Daten
Swapping: High/Low vertauschen

OK | Abbrechen | Hilfe

Wird ein Mapping, wie oben dargestellt, definiert, so wird der gesamte DPM-Inhalt (8k, ab Offset 0) als Register 1 bis 4096 interpretiert. Erst dann wenn eine ModBus-Transaktion ausgeführt wird durchsucht der PDnetIP-Controller die Mappings um die ModBus-Transaktion bearbeiten zu können.

Achtung! Liegt für eine ModBus-Transaktion kein Mapping vor wird die Transaktion mit einer Fehlermeldung beendet.

Für einen Speicherbereich können mehrere Mappings definiert werden. So ist es möglich, für den gesamten Signalspeicher Mappings mit der Eigenschaft „Read-Only“ zu definieren. Ergänzt um Mappings ohne die Eigenschaft „Read-Only“ für die auch zu schreibenden Daten, kann der andere Signalspeicher vor externen Zugriffen geschützt werden.

Die Eigenschaft „Swapping“ führt zu einer datentypgerechten Vertauschung von HI und LO-Information. Damit können im allgemeinen Probleme zwischen Big und Little-Endian Systemen vermieden werden.

Achtung! Eine ModBus-Transaktion, die sich auf Elemente bezieht, die nicht vollständig einem Mapping zugeordnet werden können, kann nicht bearbeitet werden.

4.3.2.2 ModBus Server

Der ModBus Server ermöglicht den externen Zugriff auf die via Mapping freigegebenen Bereiche des Signalspeicher. Dazu genügt es, in NetPro den ModBus-Dienst mit den ModBus Mappings (siehe Kapitel 4.3.2.1) anzulegen. Die nachfolgende Tabelle gibt einen Überblick der im Server-Betrieb unterstützten ModBus/TCP-Funktionen:

- 01 (0x01) Read Coils
- 03 (0x03) Read Holding Registers
- 05 (0x05) Write Single Coil
- 06 (0x06) Write Single Register
- 07 (0x07) Read Exception State
- 08 (0x08) Diagnostic (Echo Query, Get Statistic)
- 15 (0x0F) Write Multiple Coils
- 16 (0x10) Write Multiple Registers

4.3.2.3 ModBus Master

Der ModBus Master wird im PAD-800e über -Auftragsblöcke gesteuert. Ein Auftragsblock ist ein Bereich von 16 zusammenhängenden Merkerworten im Signalspeicher der Steuerung. Die Startadressen aller Auftragsblöcke werden in NetPro definiert. Die nachfolgende Tabelle gibt eine Übersicht über den Aufbau eines Auftragsblocks.

Wort	Kürzel	Bedeutung
1	Statuswort	Aus dem Statuswort geht der aktuelle Zustand des Auftragsblockes hervor. **1 = IDLE** Der Auftragspuffer ist frei. Dieser Status ist für die Auftragsverwaltung der SPS reserviert und wird vom PDnetIP-Controller nicht benötigt. Der Status muss nicht genutzt werden. **2 = START** Dieser Status übergibt die Kontrolle an den PDnetIP-Controller. Zu diesem Zeitpunkt muss der gesamte Auftragsblock vollständig mit den richtigen Daten gefüllt sein. Das Anwenderprogramm der SPS darf die Werte solange nicht verändern, bis der PDnetIP-Controller die Verarbeitung des Auftragsblockes abgeschlossen hat. Dies wird durch den Status 4 signalisiert. **3 = RUN** Über diesen Status zeigt der PDnetIP-Controller, dass der Auftrag erkannt wurde. Während der gesamten Verarbeitung steht dieser Status an. **4 = DONE** Der PDnetIP-Controller hat den Auftrag abgeschlossen und das Ergebnis des Auftrages kann dem Fehlerwort (Wort 15) entnommen werden. Der Auftragsblock steht dem Anwenderprogramm wieder zur Verfügung.
2	Funktionscode	**1 = Read Register** Der PDnetIP-Controller liest aus einer anderen ModBus/TCP Station Daten aus dem Register-Bereich und legt diese im gemappten Register-Bereich der A350/A500 ab. **2 = Write Register** Der PDnetIP-Controller schreibt Daten aus dem gemappten Register-Bereich in den Register-Bereich einer anderen ModBus/TCP Station. **3 = Read Coils** Der PDnetIP-Controller liest aus einer anderen ModBus/TCP Station Daten aus dem Coil-Bereich und legt diese im gemappten Coil-Bereich der A350/A500 ab. **4 = Write Coils** Der PDnetIP-Controller schreibt Daten aus dem gemappten Coil-Bereich der A350/A500 in den Coil-Bereich einer anderen ModBus/TCP Station.
3	Datentyp	Reserviert
4	ErwAdresse	Reserviert

Wort	Kürzel	Bedeutung
5	Lokale-Addresse	Adresse des zu transferierenden Signalspeichers in der eigenen (lokalen) Steuerung. Bei Funktionscode 1+2 ist es eine Register-Adresse und bei Funktionscode 3+4 eine Coil-Adresse. Für die hier angegebene Adresse muss zuvor mit NetPro ein Mapping definiert werden.
6	Anzahl	Anzahl zu bearbeitender Daten. Der Wert bestimmt somit die Anzahl zu lesender/schreibender Daten im Signalspeicher.
7	IP-Adresse 1	IP-Adresse der (anderen) ModBus/TCP Station
8	IP-Adresse 2	
9	IP-Adresse 3	
10	IP-Adresse 4	
11	UnitID	Unit-ID (ModBus/TCP Daten) - Die UnitID wird im ModBus/TCP-Telegramm abgelegt und ist eventuell für die weitere Verarbeitung in der (anderen) ModBus/TCP Station relevant. Dies betrifft jedoch nur Geräte mit Routing-Fähigkeit und muss der jeweiligen Gerätedokumentation der Partnerstation entnommen werden.
12	TRID	Transaktions-ID (ModBus/TCP Daten) - Die TRID wird im ModBus/TCP Telegramm abgelegt und hat für die weitere Verarbeitung im PDnetIP-Controller keine Bedeutung. Über die Transaktion-ID kann ein ModBus/TCP Telegramm markiert werden, so dass es sich mit einem Netzwerk-Sniffer eindeutig identifizieren lässt. Dazu ist die Transaktions-ID vor jeder Beauftragung am einfachsten zu inkrementieren.
13	Entfernte Adresse	Registeradresse des anderen (Remote) ModBus/TCP Gerätes. Diese Adresse bestimmt den Speicherbereich (Register/Coils) auf den sich die Operation (Read/Write) bezieht.
14	Timeout	Reserviert
15	Fehlerwort	Nach Verarbeitung des Auftrages steht in diesem Wort der Abschlussstatus der Operation. Die Tabelle „Fehlercodes“ dokumentiert die hier möglichen Werte.
16	Info	Steht im Fehlerwort der Fehlercode 112 (Die Partnerstation hat den Auftrag mit einer Exception beantwortet), dann beinhaltet dieses Wort den in der Exception übertragenen „Exception Code“.

Die nachfolgende Tabelle zeigt wie der ModBus-Auftragsblock vom Anwenderprogramm der SPS aus zu steuern ist.

Ablauf im Auftragsblock

Schritt	Status-wort	
1	**1 oder 4**	Der Auftragsblock wird nicht vom PDnetIP-Controller bearbeitet und kann daher vom Anwenderprogramm der SPS geändert werden.
2	**2**	Das Anwenderprogramm der SPS hat den Auftragsblock mit Daten gefüllt und startet durch Setzen des Statuswort auf 2 den Bearbeitungsvorgang.
3	**3**	Der PDnetIP-Controller bearbeitet den Auftragsblock und signalisiert dies über den Status 3. Der Status 3 ist nur sichtbar, wenn das Ergebnis nicht sofort verfügbar ist.
4	**4**	Die Bearbeitung des Auftragsblockes ist abgeschlossen. Über das Wort 15 (Fehlerwort) kann das Anwenderprogramm den Erfolg prüfen.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der in Wort 2 des ModBus-Auftragsblockes übergebene Funktionscode entspricht nicht dem ModBus-Funktionscode. Die nachfolgende Tabelle gibt eine Übersicht, auf welchen ModBus-Funktionscode der im Auftragsblock verwendete Funktionscode abgebildet wird.

Wort 2 Funktionscode im ModBus-Auftragsblock	Modbus-Funktionscode
1 = Read Register	03 (0x03) Read Holding Registers
2 = Write Register	16 (0x10) Write Multiple Registers
3 = Read Coil	01 (0x01) Read Coils
4 = Write Coil	15 (0x0F) Write Multiple Coils

Liste der Fehlercodes in Wort 15

Fehlercode	Bedeutung
0	Kein Fehler
100	Ungültiger Funktionscode im Auftragspuffer übergeben.
101	Es konnte keine Verbindung zur ModBus/TCP Station aufgebaut werden.
104	Für die im Auftragsblock übergebene lokale Adresse wurde kein Mapping definiert (Funktionscode 1+2: Register, Funktionscode 3+4: Coil)
105	Die übergebene UnitID ist größer als 255.
106	Die übergebene entfernte Adresse ist Null.
107	Die Anzahl der Elemente (Wort 6) ist bei dem gewählten Funktionscode nicht zulässig.
108	Die Kombination aus entfernte Adresse und Anzahl Elemente ist bei dem gewählten Funktionscode nicht zulässig.
109	Die Sendedaten konnten nicht an die Partnerstation übermittelt werden. Vermutlich wurde die Verbindung von der Partnerstation geschlossen.
110	Der Auftrag wurde an die Partnerstation übermittelt und während der PDnetIP-Controller auf die Antwort gewartet hat, wurde die Verbindung geschlossen. Manche Geräte zeigen durch das Schließen der Verbindung einen Fehlerzustand an.
111	Die empfangene Antwort hat einen falschen Funktionscode.
112	Die Partnerstation hat den Auftrag mit einer Exception beantwortet. Siehe nachfolgende Tabelle.
113	Die empfangene Antwort beinhaltet eine andere Transaktions-ID als die welche im Auftrag gesendet wurde.
114	Die Protokoll-ID im Antworttelegramm ist ungültig.
115	Die UnitID im Antworttelegramm ist ungültig.
116	Die Länge des Empfangstelegramms passt nicht zum Auftrag.
117	Die innere und äußere Länge des Empfangstelegramms passen nicht zusammen. Im ModBus-Telegramm ist die Telegrammlänge redundant abgelegt, so dass geprüft werden kann, ob diese zueinander passen.
118	Interner Fehler (omb_send)
119	Der Auftrag konnte nicht rechtzeitig abgeschlossen werden (Timeout).
120	Das Statuswort wurde normiert, da es vom Anwenderprogramm der SPS in nicht zulässiger Art geändert wurde.
121	Elemente der IP-Adresse im Auftragsblock sind größer als 255.
122	Die übergebene IP-Adresse ist ungültig.
123	Interner Fehler (VDM_DcToPtr)
124	Die in der Quittung enthaltene Adresse passt nicht zum Auftrag.
125	Die in der Quittung enthaltene Größe passt nicht zum Auftrag.

ModBus Exception Codes

Exception-Code	Bedeutung
01 = ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the server (or slave). This may be because the function code is only applicable to newer devices, and was not implemented in the unit selected. It could also indicate that the server (or slave) is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.
02 = ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the server (or slave). More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, a request with offset 96 and length 4 would succeed, a request with offset 96 and length 5 will generate exception 02.
03 = ILLEGAL DATA VALUE	A value contained in the query data field is not an allowable value for server (or slave). This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the MODBUS protocol is unaware of the significance of any particular value of any particular register.
04 = SLAVE DEVICE FAILURE	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
05 = ACKNOWLEDGE	Specialized use in conjunction with programming commands. The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the client (or master). The client (or master) can next issue a Poll Program Complete message to determine if processing is completed.
06 = SLAVE DEVICE BUSY	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long–duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
08 = MEMORY PARITY ERROR	Specialized use in conjunction with function codes 20 and 21 and reference type 6, to indicate that the extended file area failed to pass a consistency check. The server (or slave) attempted to read record file, but detected a parity error in the memory. The client (or master) can retry the request, but service may be required on the server (or slave) device.
10 (0x0A) = GATEWAY PATH UNAVAILABLE	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request. Usually means that the gateway is misconfigured or overloaded.
11 (0x0B) = GATEWAY TARGET DEVICE FAILED TO RESPOND	Specialized use in conjunction with gateways, indicates that no response was obtained from the target device. Usually means that the device is not present on the network.

Quelle: MODBUS Application Protocol Specification V1.1a

4.3.3 RFC1006 (ISO TP0 over TCP)

Der PDnetIP-Controller unterstützt **optional** RFC1006 bzw. „ISO TP0 over TCP“. RFC1006 ermöglicht eine Kommunikation von zusammenhängenden Datenbereichen zu anderen RFC1006-Partnerstationen.

Auf Netzebene werden die Daten gemäß ISO 8073 (TP0) über ein TCP/IP Netzwerk in Blöcken übertragen. Der PDnetIP-Controller überträgt die Netto-Daten direkt in den Signalspeicher der Steuerung.

Bevor RFC1006 genutzt werden kann, muss mit NetPro eine entsprechende Projektierung angelegt werden. Dabei werden einzelne Verbindungen definiert, welche über fest zugeordnete Sende- und Empfangspuffer verfügen. Jede Verbindung verfügt über eine Auftragsart SEND, RECEIVE oder SEND und RECEIVE. Eine Verbindung mit der Auftragsart SEND kann nur Daten senden und keine Daten empfangen. Entsprechend kann eine Verbindung mit der Auftragsart RECEIVE nur Daten empfangen und keine Daten senden. Die Betriebsart SEND/RECEIVE ermöglicht eine bidirektionale Kommunikation.

Die in jeder Verbindung definierten TSAP´s können eine Länge von 2 bis 16 Zeichen besitzen.

Zur Steuerung der Sende- und Empfangsvorgänge werden ein Merker- und Merkerwortbereich benötigt. Die Startadressen beider Speicherbereiche werden mit NetPro festgelegt. Dabei erzeugt NetPro, bei Bedarf eine S-Liste oder ASD-Datei, in welcher die in den Speicherbereichen automatisch definierten Elemente aufgeführt werden.

Achtung! Nach jeder Veränderung der RFC1006 Projektierung (Verbindung einfügen oder löschen) kann sich die Belegung des Signalspeichers verändern.

Achtung! Wird die Auftragsart von SEND/RECEIVE in SEND oder RECEIVE geändert, ändert sich die Belegung des Signalspeichers.

Achtung! Nach Änderungen an der Signalspeicherbelegung kann es notwendig sein, das Anwenderprogramm der SPS anzupassen. Es ist immer sinnvoll, die Signalspeicherbelegung mit NetPro generieren zu lassen und nicht manuell zu erzeugen. Die von NetPro erzeugte S-Liste oder ASD-Datei kann leicht in den jeweiligen Programmiersystemen genutzt werden.

Die maximale Größe einer TPDU´s beträgt 2.048 Byte. Baut der PDnetIP-Controller aktiv eine Verbindung auf, so wird im „Connection Request“ eine TPDU-Größe von 2.048 Byte festgelegt. Antwortet die Partnerstation im „Connection Confirm“ mit einer kleineren TPDU-Größe, nutzt der PDnetIP-Controller diese.

Baut eine andere Station aktiv eine Verbindung zum PDnetIP-Controller auf, nutzt dieser die im „Connection Request“ übermittelte TPDU-Größe, wenn diese nicht größer als 2.048 Byte ist. Andernfalls quittiert der PDnetIP-Controller im „Connection Confirm“ die TPDU-Größe mit 2.048 Byte und erwartet, dass die Partnerstation diese Größe in den Sendedaten benutzt.

Die folgende Tabelle beschreibt die Bedeutung der für jede Verbindung generierten Status-Worte und Merker. Der Suffix ergänzt den mit NetPro projektierten Verbindungsnamen.

Vorhanden	Suffix	Zustand
Immer	'_ID'	Die mit NetPro projektierte Verbindungskennung.
	'_STATE'	Der aktuelle Verbindungsstatus.
Senderichtung	'_SB'	Über das Sendebit _SB zeigt das Anwenderprogramm der SPS dem PDnetIP-Controller an, dass ein Sendetelegramm vorliegt.
	'_SQ'	Über das Quittungsbit wird das Anwenderprogramm vom PDnetIP-Controller über das Ende des Sendevorganges informiert.
	'_AWS'	Letzter/Aktueller Status in Senderichtung.
Empfangsrichtung	'_EB'	Über das Empfangsbit _SQ informiert der PDnetIP-Controller das Anwenderprogramm der SPS über ein neues Empfangstelegramm.
	'_EQ'	Mit dem Quittungsbit informiert das Anwenderprogramm den PDnetIP-Controller über die Verarbeitung der Empfangsdaten.
	'_AWE'	Letzter/Aktueller Status in Empfangsrichtung.

Detaillierte Informationen zum Ablauf des Sendevorgang befinden sich im Kapitel 4.3.3.2 und zum Empfangsvorgang in Kapitel 4.3.3.3.

4.3.3.1 Verbindungsstatus STATE

Die nachfolgende Tabelle dokumentiert alle möglichen Werte für den RFC1006-Verbindungsstatus.

Wert	Bedeutung
1	Verbindung wurde von der Partner-Station geöffnet. Dieser Status liegt vor, wenn die Verbindung im PDnetIP-Controller als passiv projektiert wurde.
2	Der PDnetIP-Controller baut eine Verbindung auf. Der Status steht solange an, bis der eigene Connection Request (CR) mit einem Connection Confiorm (CC) der Partnerstation bestätigt wurde.
3	Der PDnetIP-Controller besitzt eine aktive Verbindung zur Partnerstation. Diese Verbindung wurde aktiv aufgebaut.
4	Es besteht keine Verbindung zu einer Partnerstation.

4.3.3.2 Sendevorgang und Sendebits SB/SQ

Mögliche Zustände SB/SQ

SB	SQ	Zustand
0	1	Sendepuffer frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der SB/SQ-Merker durchgeführt werden muss.
1	0	Dieser Zustand wird vom Anwenderprogramm der SPS gesetzt, wenn ein Telegramm gesendet werden soll und der Sendepuffer mit den Sendedaten befüllt wurde. Wurde der Sendevorgang angestoßen darf das Anwenderprogramm die Merker SB/SQ nicht mehr verändern. Das Ende des Sendevorganges wird über SB/SQ angezeigt. Das Ergebnis des Sendevorganges wird über AWS angezeigt. Der Sendepuffer darf bis zum Abschluss des Sendevorganges nicht verändert werden.
0	0	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller auf SB=0 und SQ=1 normiert.
1	1	

Ablauf Sendevorgang

Schritt	SB	SQ	Zustand
1	0	1	Der Sendepuffer ist verfügbar.
2	1	0	Das Anwenderprogramm der SPS hat den Sendepuffer mit Daten gefüllt und den Sendevorgang gestartet.
3	0	1	Der PDnetIP-Controller hat den Sendeauftrag bearbeitet und der Sendepuffer steht wieder zur Verfügung.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

4.3.3.3 Empfangsvorgang und Empfangsbits EB/EQ

Mögliche Zustände EB/EQ

EB	EQ	Zustand
0	0	Der Empfangspuffer ist frei. Dieser Zustand wird vom PDnetIP-Controller im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker durchgeführt werden muss.
0	1	Zustand ist nicht zugelassen und wird vom PDnetIP-Controller normiert.
1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss vom Anwenderprogramm der SPS quittiert werden. Erfolgt dies nicht (SPS wird angehalten), wird das Telegramm über ein TimeOut gelöscht.
1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und zeigt dies über das Setzen von EQ dem PDnetIP-Controller an. Es ist wichtig das vom Anwenderprogramm ausschließlich das EQ-Bit gesetzt wird.

Ablauf Empfangsvorgang

Schritt	EB	EQ	Zustand
1	0	0	Empfangspuffer ist frei
2	1	0	Es wurde ein Telegramm empfangen
3	1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet
4	0	0	Empfangspuffer ist wieder frei

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

4.3.3.4 Sende-/Empfangsstatus AWS/AWE

Fehlerliste Ausgangsworte (_AWS und _AWE)

Wert	Bedeutung
0	Kein Fehler
104	Verbindung wurde geschlossen
105	Aktiver Verbindungsaufbau gescheitert
106	Es ist keine Verbindung vorhanden. Die Sendedaten können nicht zugestellt werden, da der PDnetIP-Controller selbst keine Verbindung aufbauen darf. Dabei handelt es sich um eine Konfigurationseigenschaft der Verbindung.
107	Sendevorgang gescheitert
109	Die Zahl gleichzeitig ausführbarer Sendevorgänge von 12 PDU´s pro Verbindung wurde überschritten.
111	TimeOut Sendedaten
112	TimeOut Empfangsdaten
113	Empfangsdaten konnten nicht verarbeitet werden, da die Struktur der Empfangsdaten zuvor nicht registriert wurde. Strukturen müssen über den Baustein PDSTR beim PDnetIP-Controller registriert werden.
114	Sendedaten konnten nicht verarbeitet werden, da die Struktur der Sendedaten zuvor nicht registriert wurde. Strukturen werden mit dem Baustein PDSTR beim PDnetIP-Controller registriert.

4.3.4 TCP-Schnittstelle

Der PDnetIP-Controller unterstützt **optional** eine TCP-Schnittstelle die einen direkten Austausch der auf dem TCP-Layer gesendeten und empfangenen Daten mit dem Anwenderprogramm der SPS ermöglicht.

Bevor die freie TCP-Schnittstelle genutzt werden kann, muss mit NetPro eine entsprechende Projektierung angelegt werden. Dabei werden einzelne Verbindungen definiert, welche über fest zugeordnete Sende- und Empfangspuffer verfügen. Jede Verbindung verfügt über einen Auftragsart SEND, RECEIVE oder SEND und RECEIVE. Eine Verbindung mit der Auftragsart SEND kann nur Daten senden und keine Daten empfangen. Entsprechend kann eine Verbindung mit der Auftragsart RECEIVE nur Daten empfangen und keine Daten senden. Die Betriebsart SEND/RECEIVE ermöglicht eine bidirektionale Kommunikation.

Zur Steuerung der Sende- und Empfangsvorgänge werden zwei jeweils zusammenhängende Speicherbereiche benötigt. Ein Speicherbereich liegt im Merkerbereich und ein weiterer im Merkerwortbereich. Mit NetPro wird für beide Speicherbereiche die Startadresse festgelegt. Dabei erzeugt NetPro optional eine S-Liste oder ASD-Datei, in welcher die in den Speicherbereichen automatisch definierten Elemente aufgeführt werden.

Achtung! Nach jeder Veränderung der Projektierung (Verbindung einfügen oder löschen) kann sich die Belegung des Signalspeichers verändern. Es ist immer sinnvoll, die Signalspeicherbelegung mit NetPro generieren zu lassen und nicht manuell zu erzeugen. Die von NetPro erzeugte S-Liste oder ASD-Datei kann leicht in den jeweiligen Programmiersystemen genutzt werden.

Die Gesamtgröße der Auftragspuffer darf den zur Verfügung stehenden Arbeitsspeicher des PDnetIP-Controllers nicht überschreiten. Ein Auftragspuffer sollte die Größe von 4.096 Byte nicht überschreiten.

Alle gesendeten/empfangenen Daten werden vom PDnetIP-Controller ohne Veränderung dem Sendepuffer entnommen bzw. im Empfangspuffer abgelegt.

Eine Manipulation der Byte-Order erfolgt nicht, alle Daten verbleiben in ihrem Format. Dies bedeutet, dass die Daten zwischen Socket und Signalspeicher 1:1 kopiert werden.

Wird eine Verbindung zu einer Port-Nummer aufgebaut zu der bereits eine Verbindung besteht, wird die vorhandene Verbindung geschlossen. Zu einem Zeitpunkt kann pro Portnummer nur eine aktive Verbindung vorhanden sein.

Achtung! Nach Änderungen an der Signalspeicherbelegung kann es notwendig sein, das Anwenderprogramm der SPS anzupassen.

Die folgende Tabelle beschreibt die Bedeutung der für jede Verbindung belegten Status-Worte und Merker. Der Suffix ergänzt den mit NetPro projektierten Verbindungsnamen.

Vorhanden	Suffix	Zustand
Immer	'_ID'	Die mit NetPro projektierte Verbindungskennung.
	'_STATE'	Der aktuelle Verbindungsstatus.
Sende-richtung	'_SB'	Über das Sendebit _SB zeigt das Anwenderprogramm der SPS dem PDnetIP-Controller an, dass ein Sendetelegramm vorliegt.
	'_SQ'	Über das Quittungsbit wird das Anwenderprogramm vom PDnetIP-Controller über das Ende des Sendevorganges informiert.
	'_TXSTATE'	Letzter/Aktueller Status in Senderichtung.
	,_TXLEN'	Anzahl zu sendender Bytes
	,_TXSTRUC'	Kennung der zu sendenden Struktur *)
Empfangs-richtung	'_EB'	Über das Empfangsbit _SQ informiert der PDnetIP-Controller das Anwenderprogramm der SPS über ein neues Empfangstelegramm.
	'_EQ'	Mit dem Quittungsbit informiert das Anwenderprogramm den PDnetIP-Controller über die Verarbeitung der Empfangsdaten.
	'_RXSTATE'	Letzter/Aktueller Status in Empfangsrichtung.
	,_RXLEN'	Anzahl empfangener Bytes
	,_RXSTRUC'	Kennung der empfangenen Struktur *)

*) Diese Worte werden nur in der Übertragungsart „Header mit Kennung" genutzt, in diesem Fall bezieht der PDnetIP-Controller die Längeninformationen weder aus RXLEN und TXLEN, noch aus den projektierten Strukturen.

Detaillierte Informationen zum Ablauf des Sendevorgang befinden sich im Kapitel 4.3.4.2 und zum Empfangsvorgang in Kapitel 4.3.4.4.

4.3.4.1 Verbindungsstatus STATE

Die nachfolgende Tabelle dokumentiert alle möglichen Werte für den TCP-Verbindungsstatus.

Wert	Bedeutung
1	Es besteht keine Verbindung zu einer Partnerstation.
2	Der PDnetIP-Controller baut eine Verbindung zur Partnerstation auf.
3	Der PDnetIP-Controller besitzt eine aktive Verbindung zur Partnerstation.
9	Verbindungsstatus unbekannt

4.3.4.2 Sendevorgang und Sendebits SB/SQ

Mögliche Zustände SB/SQ

SB	SQ	Zustand
0	1	Sendepuffer frei. Dieser Zustand wird vom PDnetIP-Controller auch im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der SB/SQ-Merker durchgeführt werden muss.
1	0	Dieser Zustand wird vom Anwenderprogramm der SPS gesetzt, wenn ein Telegramm gesendet werden soll und der Sendepuffer mit den Sendedaten befüllt wurde. Wurde der Sendevorgang angestoßen darf das Anwenderprogramm die Merker SB/SQ nicht mehr verändern. Das Ende des Sendevorganges wird über SB/SQ angezeigt. Das Ergebnis des Sendevorganges wird über TXSTATE angezeigt. Der Sendepuffer darf bis zum Abschluss des Sendevorganges nicht verändert werden.
0	0	Diese Zustände sind nicht zugelassen und werden vom PDnetIP-Controller normiert. Die Normierung setzt den SB auf Null und SQ auf Eins und gibt damit den Sendepuffer wieder frei.
1	1	

Ablauf Sendevorgang

Schritt	SB	SQ	Zustand
1	0	1	Der Sendepuffer ist verfügbar.
2	1	0	Das Anwenderprogramm der SPS hat den Sendepuffer mit Daten gefüllt und den Sendevorgang gestartet.
3	0	1	Der PDnetIP-Controller hat den Sendeauftrag bearbeitet und der Sendepuffer steht wieder zur Verfügung.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

Das Sendetelegramm wird nur dann übertragen, wenn bei der Partnerstation genügend Platz im Input-Window des Sockets zur Verfügung steht. Der PDnetIP-Controller arbeitet so, um eine Fragmentierung der Sendedaten zu vermeiden. Bedingt dadurch, dass bei einem „freien“ TCP-Protokoll kein Anfang/Ende eines Telegramms von der Empfangsstation erkannt werden kann, nutzt der PDnetIP-Controller das beschriebene Verhalten. Kann das Telegramm nicht übertragen werden, generiert der PDnetIP-Controller den Status 35 und beendet den Sendevorgang. Die vollständige Sendestatusliste befindet sich in Kapitel 4.3.4.3.

4.3.4.3 Sendestatus TXSTATE

Sendestatus _TXSTATE

Wert	Bedeutung
0	Kein Fehler
20	Initialisierung beim Systemstart. Dieser Status wird einmal im Einschaltmoment gesetzt
21	Sendedatenlänge ungültig Es wurde keine Länge oder eine zu große Länge in TXLEN geladen. Die maximale Länge wird in NetPro in der Projektierung des Sendepuffer festgelegt.
22	Der Sendevorgang ist gescheitert, obwohl im Output-Window der Verbindung für die Sendedaten genügend Platz vorhanden ist.
23	Der Verbindungsaufbau zur Partnerstation ist gescheitert.
24	Die Kombination von SB/SQ ist nicht zulässig. Werden SB/SQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
25	SB/SQ wurde während eines laufenden Sendevorganges vom Anwenderprogramm der SPS geändert. Werden SB/SQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
26	Der Sendepuffer kann nicht gelesen werden, da die Struktur zuvor nicht erfolgreich mit dem PDSTR beim PDnetIP-Controller registriert wurde.
27	Die Verbindung zur Partnerstation wurde geschlossen.
28	Nach dem Start des Sendevorganges durch das Anwenderprogramm (SB=1, SQ=0) überwacht der PDnetIP-Controller ob der Sendevorgang innerhalb einer TimeOut-Zeit abgeschlossen werden kann. Andernfalls wird der Sendevorgang abgebrochen. Die TimeOut-Zeit ist von der Firmware-Version abhängig und beläuft sich in der Firmware 4.10.51.01 auf ca. 10 Sekunden.
29 *)	Statusmeldung: Die Sendedaten liegen bereit und werden an die Partnerstation übertragen.
30 *)	Statusmeldung: Der PDnetIP-Controller hat den Start eines Sendevorganges durch das Anwenderprogramm der SPS erkannt
31 *)	Statusmeldung: Der PDnetIP-Controller wartet auf die Sendedatenlänge.
32 *)	Statusmeldung: Der Sendevorgang wurde gestartet und der PDnetIP-Controller wartet auf den Sendepuffer.
33 *)	Statusmeldung: Der Sendevorgang wurde gestartet und der PDnetIP-Controller wartet auf die Strukturkennung um den Sendepuffer zu bestimmen.
34	Die in Wort TXSTRUC übergebene Strukturkennung ist in der Projektierung nicht enthalten.
35	Die Kommunikation zur Partnerstation ist zu langsam. Der PDnetIP-Controller kann die Daten nicht so schnell zustellen, wie diese beauftragt werden.
36	Sendedatenlänge zu groß. Der PDnetIP-Controller kann in der Übertragungsart „Header mit Kennung“ maximal 510 Bytes senden.
37	Der Sendevorgang ist gescheitert, da keine Verbindung aufgebaut wurde und keine Partnerstation projektiert ist.
38	Optimierte Datenübertragung wurde projektiert aber der Datenpuffer ist noch ungültig.

Wert	Bedeutung
39	Es ist kein automatischer Verbindungsaufbau projektiert. Da keine Verbindung zu einer Partnerstation bestand, wurde der Sendeauftrag negativ quittiert.

*) Die markierten Werte stehen nicht beim Abschluss eines Sende-/Empfangsvorganges an und dienen der Diagnose zur Laufzeit.

4.3.4.4 Empfangsvorgang und Empfangsbits EB/EQ

Mögliche Zustände EB/EQ

EB	EQ	Zustand
0	0	Empfangspuffer frei. Dieser Zustand wird vom PDnetIP-Controller auch im Einschaltmoment gesetzt, so dass im Anwenderprogramm der SPS keine Normierung der EB/EQ-Merker notwendig ist.
0	1	Dieser Zustand ist nicht zugelassen und wird vom PDnetIP-Controller normiert.
1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt. Das Telegramm muss sofort vom Anwenderprogramm verarbeitet werden. Das Ende des Verarbeitungsvorganges wird vom Anwenderprogramm durch das setzen des EQ-Bit signalisiert. Mit dem setzen des EQ-Bit gibt das Anwenderprogramm auch die Kontrolle über den Empfangspuffer an den PDnetIP-Controller zurück. Wird das Telegramm nicht rechtzeitig vom Anwenderprogramm quittiert bricht der PDnetIP-Controller den Empfangsvorgang automatisch ab.
1	1	Das Anwenderprogramm der SPS hat das Telegramm verarbeitet und quittiert. Es ist wichtig das vom Anwenderprogramm ausschließlich das EQ-Bit gesetzt wird.

Ablauf Empfangsvorgang

Schritt	EB	EQ	Zustand
1	0	0	Empfangspuffer frei.
2	1	0	Der PDnetIP-Controller hat ein Telegramm empfangen und im Empfangspuffer abgelegt.
3	1	1	Das Anwenderprogramm hat das Telegramm verarbeitet und quittiert.
4	0	0	Der Empfangspuffer ist wieder verfügbar.

Der Zustand wird vom PDnetIP-Controller gesetzt und dient dem Anwenderprogramm der SPS als Triggerbedingung.

Der Zustand wird vom Anwenderprogramm der SPS gesetzt und dient der Firmware des PDnetIP-Controllers als Triggerbedingung.

Der Zustand ist nicht zulässig und wird von einem fehlerhaften Anwenderprogramm der SPS verursacht.

4.3.4.5 Empfangsstatus RXSTATE

Empfangsstatus _RXSTATE

Wert	Bedeutung
0	Kein Fehler
50	Initialisierung beim Systemstart. Dieser Status wird einmal im Einschaltmoment gesetzt
51	Es liegen mehr Empfangsdaten vor, als im Empfangspuffer abgelegt werden können. Es werden soviel Daten empfangen, wie in den Empfangspuffer der SPS passen. Die verbliebenden Daten werden im nächsten Empfangszyklus im Empfangspuffer übergeben.
52	Die Verbindung zur Partnerstation wurde geschlossen.
53	Der Empfangspuffer kann nicht beschrieben werden, da die empfangene Struktur zuvor nicht beim PDnetIP-Controller registriert wurde.
55	Die Kombination von EB/EQ ist nicht zulässig. Werden EB/EQ zur Laufzeit in nicht zulässiger Art geändert, kann es zu nicht vorhersehbaren Kommunikationsproblemen führen. Die Ursache für diesen Fehler muss im Anwenderprogramm der SPS abgestellt werden.
56	Beim Start des Empfangsvorganges durch den PDnetIP-Controller (Daten wurden empfangen) überwacht der PDnetIP-Controller ob der Empfangsvorgang innerhalb einer Timeout-Zeit abgeschlossen wurde. Andernfalls wird der Empfangsvorgang abgebrochen und die Empfangsdaten werden gelöscht. Die Timeout-Zeit ist von der Firmware-Version abhängig und beläuft sich in der Firmware-Version 4.10.51.01 auf ca. 10 Sekunden.
57	Statusmeldung: Der PDnetIP-Controller wartet auf das Durchschreiben der Empfangslänge RXLEN.
58	Statusmeldung: Der PDnetIP-Controller wartet auf das Durchschreiben der Empfangsdaten.
59	Die empfangene Strukturkennung ist in der Projektierung nicht enthalten. Daher ist es dem PDnetIP-Controller nicht möglich den Daten einen Empfangspuffer zuzuordnen. Die Verbindung wird deswegen geschlossen.
60	Der Empfangsvorgang für den Header ist gescheitert, obwohl im Input-Window der Verbindung die Daten verfügbar sind.
61	Der Empfangsvorgang wurde abgebrochen, da ein leerer Sendepuffer (Größe=0) konfiguriert wurde.
62	Der Empfangsvorgang ist gescheitert, obwohl im Input-Window der Verbindung Daten vorhanden sind.

4.3.4.6 Übertragungsart „Header mit Kennung“

In der Übertragungsart „Header mit Kennung“ erfolgt keine statische Zuordnung der Sende-/Empfangspuffer. Aus der in der NetPro-Projektierung des TCP-Interfaces zu erstellenden Tabelle „Strukturen“ entnimmt die Firmware alle benötigten Informationen, wo die Sende-/Empfangsdaten in der SPS liegen.

Die Tabelle „Strukturen“ beinhaltet für jeden Eintrag die folgenden Informationen:

- **Protokoll-Kennung**. Die Protokoll-Kennung ist die im Header übertragene Kennung (LB/HB), welche den Inhalt der Daten klassifiziert.
- **Struktur-Kennung**. Die Struktur-Kennung identifiziert die der Protokoll-Kennung zugeordneten Sende-/Empfangspuffer im Signalspeicher der Steuerung. Die Strukturen müssen zuvor mit dem Baustein PDSTR bei dem PDnetIP-Controller angemeldet werden.
- **Größe**. Die Größe (in Byte) spezifiziert, wie groß der über den Socket übertragende Datenbereich ist. Dadurch können mehre hintereinander gesendete Telegramme zerlegt werden.

Der PDnetIP-Controller legt für jeden Eintrag der Tabelle einen Puffer in Sende- und Empfangsrichtung an.

Wichtig: Zu einem Zeitpunkt kann der gleiche Puffer in Empfangsrichtung nur einmal genutzt werden. Dieses Verhalten muss von den Partnerstationen gewährleistet werden.

Abläufe:

- **Empfangsvorgang**
  Empfängt der PDnetIP-Controller einen Header, schlägt er die im Header enthaltene Struktur-Kennung in der Tabelle „Strukturen“ nach. Aus der Tabelle leitet sich die Adresslage des Empfangspuffers (Struktur-Kennung) ab. Das Anwenderprogramm der SPS kann dem Wort „RXSTRUC“ entnehmen, welche Struktur empfangen wurde.
- **Sendevorgang**
  Beauftragt das Anwenderprogramm der SPS einen Sendevorgang, entnimmt der PDnetIP-Controller dem Wort „TXSTRUC“ welche Struktur gesendet wird.

Die Längeninformation wird ebenfalls der Tabelle entnommen. Die Worte RXLEN und TXLEN haben in dieser Übertragungsart keine Bedeutung.

Achtung: In der Übertragungsart „Header mit Kennung“ dürfen die gesendeten Strukturen maximal 510 Byte groß sein.

Achtung! Diese Einstellung ist nur für spezielle Anlagen sinnvoll und gehört nicht zum Standardleistungsumfang des PAD-800e.

4.3.4.7 Option „Zeitoptimierte Kommunikation zum Endgerät“

Diese Einstellung hat für den PAD-800e keine Bedeutung.

4.3.4.8 Option „Beauftragung ohne Quittungsbehandlung“

In dieser Betriebsart arbeitet der PDnetIP-Controller ohne Quittungsbit und es gelten andere Regeln für die Verwendung von SB/SQ sowie EB/EQ. Um diese Betriebsart nutzen zu können,, muss die Option „Zeitoptimierte Kommunikation zum Endgerät“ aktiviert werden.

Am Anfang des SPS Zyklus müssen alle Sendebits gelöscht und am Ende des SPS Zyklus alle Empfangsbits gelöscht werden. Diese Betriebsart ermöglicht es dem PDnetIP-Controller pro SPS-Zyklus ein Telegramm zu senden. Das Anwenderprogramm der SPS bekommt keine Meldung darüber, ob ein Sendevorgang erfolgreich war.

Achtung! Diese Einstellung ist nur für spezielle Anlagen sinnvoll und gehört nicht zum Standardleistungsumfang des PAD-800e.

4.3.4.9 Option „KEEP ALIVE senden“

Wird diese Option aktiviert, sendet der PDnetIP-Controller zyklisch ein KEEP-ALIVE an die Partnerstation und die Zeitüberwachung der Verbindung auf Inaktivität der Kommunikation wird deaktiviert. Eine in dieser Betriebsart geöffnete Verbindung kann somit solange geöffnet bleiben, wie die Partnerstation erreichbar ist. Andernfalls baut der PDnetIP-Controller nach einer bestimmten Zeit eine nicht genutzte Verbindung ab.

4.3.5 VDM

Jeder PDnetIP-Controller unterstützt die Kommunikation über das VDM-Modul. Der VDM belastet den Anwender nicht mit der Implementierung von Kommunikationsaufgaben. Es genügt die zu koppelnden Daten mit NetPro zu projektieren. Informationen zur Projektierung des VDM und somit der Datenzellen befinden sich im Handbuch von NetPro.

4.3.6 SYSLOG

Der PDnetIP-Controller unterstützt die Kommunikation mit einem Syslog-Server. Um den Syslog des PDnetIP-Controllers zu aktivieren, muss auf einer Station vom NetPro-Typ „Extern“ oder vom NetPro-Typ „PDM-Server“ ein Syslog-Server laufen. Fehlt im NetPro-Projekt diese Station, auf welcher der Syslog-Server läuft, muss diese mit dem NetPro-Typ „Extern“ eingefügt werden. Bei allen PDnetIP-Controllern,welche Syslog-Meldungen generieren, ist dann in der Projektierung des Syslog unter PDnetIP/Dienste/Syslog diese Station auszuwählen. Der Einsatz des Syslog vereinfacht die Diagnose entscheidend.

Schritt 1) Syslog unter PDnetIP/Dienste aktivieren

Zentrale / PDnetIP/PDnetIP

IP-Adressierung | Dienste | Einstellungen

- [ ] ModBus/TCP
- [ ] RFC 1006
- [ ] TCP
- [x] Syslog
- [ ] SNTP

Bearbeiten

OK | Abbrechen | Hilfe

Schritt 2) Zielstation der Syslog-Meldungen auswählen

Syslog

Syslog-Station SYSLOG-Server (192.168.3.111) ...

OK Abbrechen

Achtung! Je nach Anlage kann bereits ein projektierter SYSLOG-Server die Performance reduzieren. Dies ist nicht der Fall wenn die Station arbeitet. Werden jedoch Fehler gemeldet, Verbindungen auf- und abgebaut benötigt das Diagnose-Modul Ressourcen, welche sonst den anderen Modulen zur Verfügung gestanden hätten.

4.4 Serielle Protokolle

Der PDnetIP-Controller unterstützt eine Reihe serieller Protokolle zur Kommunikation mit anderen Geräten über offene Schnittstellen sowie kundenspezifische Implementierungen als Sonderlösung.

4.4.1 Access-Server

Der AccessServer ermöglicht einen Fernanschluss einer seriellen Schnittstelle. Dazu empfängt und sendet der PDnetIP-Controller Daten über die serielle Schnittstelle stellvertretend für ein anderes Gerät. Das Gerät, welches diese Funktionalität nutzen möchte, muss dazu eine TCP-Verbindung zum PDnetIP-Controller aufbauen.

Das Profil „A800 Programmiergerät“ wählt die für die Funktionen Fernbedienen und Fernladen notwendigen Parameter. Alternativ kann das Profil „RAW-Daten“ gewählt werden.

Für die Nutzung des AccessServer wird eine Freischaltung benötigt.

4.4.2 IS-Tester

Der Einsatz des IS-Tester in der A800 ist nur eingeschränkt möglich. Daher erfolgt an dieser Stelle keine Beschreibung des Treibers.

4.4.3 Seab 1/F Master

Der PDnetIP-Controller unterstützt **optional** das „Seab 1/F Master“ Protokoll. Dazu benötigt er mit Hilfe des PDNIP-Bausteins Zugriff auf den Signalspeicher der Steuerung.

Das Protokoll „Seab 1/F Master“ ermöglicht die Kommunikation zu folgenden Stationstypen und dazu Kompatiblen:

- Micro
- U120
- U250

Eine genaue Beschreibung erfolgt im „Seab 1/F Benutzerhandbuch für PDnetIP-Controller“. Im folgenden werden die Projektierungseinstellung sowie Leistungsparameter der Implementierung kurz dargestellt.

Mit der **lizenzpflichtigen** Vollversion von NetPro werden die Parameter des Treibers und das Datenmodell projektiert.

- **TV, Vorlaufzeit.** Zeitspanne, die nach der Aktivierung von RTS abläuft, bevor das erste Zeichen des Telegramms gesendet wird.
- **TN, Nachlaufzeit.** Zeitspanne, die nach dem Senden des letzten Zeichens abläuft, bevor RTS deaktiviert wird.
- **TP, Pausenzeit.** Zeitspanne nach Deaktivieren von RTS, in der die Stationen und deren Modems die Umschaltung zwischen Senden/Empfangen durchführen.
- **TWM, Wartezeit Master.** Antwortet die Unterstation nicht innerhalb von TWM, wird damit dem Master ein Fehler angezeigt. TWM muss größer als TP+TV sein, damit eine Unterstation ein gültiges Telegramm übertragen kann. Um eine zuverlässige Zustandserkennung zu ermöglichen, müssen TP, TV und TN bei Master und Unterstation identisch sein.
- **TVM, Verzögerungszeit Master.** Wurde vom Master ein fehlerhaftes Telegramm empfangen, so verzögert der Master den nächsten Sendevorgang um TVM, um der Unterstation den Fehler anzuzeigen.

- **Wiederholung im Fehlerfall.** Wird ein Pollaufruf nicht oder fehlerhaft beantwortet, so wird der Pollaufruf so oft wiederholt, bis eine gültige Antwort eintrifft. Mit der Wiederholungszahl wird dieser Vorgang auf eine maximale Anzahl Pollaufrufe beschränkt. Sinnvolle Werte liegen zwischen 0-3.

Für Unterstationen sind die A-Bytes 1 bis 126 und 127 gültig, wobei die virtuelle Unterstation 127 allein dem Senden von Broadcast an alle Unterstationen dient.

In die Poll-Liste können maximal 126 Unterstationen eingetragen werden, die vom Seab1F-Treiber in dieser Reihenfolge gepollt werden. Die Poll-Liste ist unsortiert, Unterstationen können mehrfach vorhanden sein.

Micro

Objekt	Anzahl / Datentyp
Meldungen	128 / Bit
Echtzeit Meldungen	keine
Echtzeit Meldungen aus dem Ringpuffer	keine
Echtzeit Meldungen aus dem Datenmodell	keine
Zählwerte	24 / Word
Umgespeicherte Zählwerte	24 / Word
Messwerte 8Bit	24 / Byte
Messwerte 16Bit	24 / Word
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	64 / Bit
Sollwerte	32 / Int. / Word (digital / analog)
Aufrufe	4 / Byte (APEX Datentyp)

U120

Objekt	Anzahl / Datentyp
Meldungen	256 / Bit
Echtzeit Meldungen	256 / Bit
Echtzeit Meldungen aus dem Ringpuffer	256 / Bit
Echtzeit Meldungen aus dem Datenmodell	256/ Bit
Zählwerte	63 / Word
Umgespeicherte Zählwerte	63 / Word
Messwerte 8Bit	64 / Byte
Messwerte 11/16Bit	63 / (11 Bit + Vz) Word
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	256 / Bit
Sollwerte	16/32 / Int./Word (digital/analog)
Aufrufe	4 / Byte (APEX Datentyp)

U250

Objekt	Anzahl / Datentyp
Meldungen	1024 / Bit (in Summe mit E. Meldungen)
Echtzeit Meldungen	1024 / Bit (in Summe mit Meldungen)
Echtzeit Meldungen aus dem Ringpuffer	1024 / Bit
Echtzeit Meldungen aus dem Datenmodell	1024 / Bit
Zählwerte	128 / Word
Umgespeicherte Zählwerte	128 / Word
Messwerte 8Bit	256 / Byte (in Summe mit 16Bit Mw.)
Messwerte 16Bit	256 / Byte (in Summe mit 8Bit Mw.)
Verwaltungsmeldungen	48 / Bit
Stationsmeldungen	9 / Byte (APEX Datentyp)
Befehle	512 / Bit Feld
Sollwerte	256 / Int. / Word (digital / analog)
Aufrufe	4 / Byte (APEX Datentyp)

Broadcast

Objekt	Anzahl / Datentyp
Aufrufe	4 / Byte (APEX Datentyp)

4.5 Kundenspezifische Protokolle

Es sind in der Firmware des PDnetIP-Controllers weitere kundenspezifische Treiber implementiert. Die entsprechende Dokumentation erfolgt dann in den kunden- bzw. projektspezifischen Dokumenten.

Benötigen Sie ein spezielles Protokoll oder eine Erweiterung einer vorhandenen Implementierung dann sprechen Sie uns darauf an.

5 Inbetriebnahme

5.1 Hardware

Der PDnetIP-Controller PAD-800e wird in einen PMB-Steckplatz einer A800-Grundmagazines eingesetzt. Beim Einstecken in das Magazin muss die Baugruppe richtig einrasten. (Keine Gewalt anwenden!) Die Baugruppe darf nur gesteckt werden, wenn die SPS ausgeschaltet ist. Nach Einrasten der Baugruppe müssen die Sicherungsschrauben angezogen werden.

5.1.1 PMB

Nachfolgend finden Sie die Informationen zur Konfiguration des PAD-800e am PMB.

5.1.1.1 Größe

Je nach Hardware-Revision stellt der PAD-800e der A800 einen Bereich von 4k oder 8k zur Verfügung. Die folgende Tabelle zeigt die möglichen Konfiguration und dafür notwendigen Größen.

ICOS	DPM-Direkt	Größe
Nein	Ja	4k oder 8k
KP-92	Nein	8k
KOS-881	Ja	8k
KOS-881	Nein	4k

5.1.1.2 Adresse

Die PMB-Adresse des PDnetIP-Controllers PAD-800e wird durch DIP-Schalter festgelegt. Vor der Kodierung der Adresslage ist es wichtig, die Konfiguration der vorhandenen Steuerung zu prüfen. Der eingestellte Speicherbereich auf dem PMB-Bus muss frei sein! Eine Übersicht aller möglichen Konfigurationsvarianten befindet sich in Kapitel 8.2.

Achtung! Die Konfiguration einer Adresseinstellung erfolgt je nach Hardware-Revision verschieden. Wird ein vorhandener PAD-800e ersetzt, kann die Einstellung der DIP-Schalter nur bei gleicher Hardware-Revision 1:1 übertragen werden.

5.2 Software

Die nachfolgenden Kapitel beschreiben die Konfiguration und Integration des PAD-800e seitens der Software.

5.2.1 Abfrage der aktuellen Firmware-Version

Welche Firmware-Version eingesetzt wird, können Sie mit DLSE32 bestimmen. Starten Sie dazu den PDnetIP-Controller im Lade-Modus und führen Sie die Funktion „`Online`“ / „`Diagnose`“ und „`Firmwareversion/Konfiguration auslesen`“ aus. Verfügt der PDnetIP-Controller bereits über eine IP-Adresse kann die Firmware-Version am einfachsten mit NetPro über das Netzwerk ermittelt werden.

Alternativ kann bei neueren Firmware-Versionen die Versionsinformation über die serielle Schnittstelle 2 bestimmt werden. Der PDnetIP-Controller gibt die Versionserkennung im Einschaltmoment aus. Mit einem üblichen Terminalprogramm kann die Versionsinformation aufgezeichnet werden. Dazu muss das Terminalprogramm die folgenden Schnittstellenparameter benutzen:

- 115.200 Baud bei einem PDnetIP-Controller 33 MHz oder 19.200 Baud bei einem PDnetIP-Controller mit 25 MHz
- 8 Datenbits
- 1 Stopbit
- keine Parität

5.2.2 Abfrage der aktuellen Lader-Version

Welche Lader-Version eingesetzt wird, können Sie mit DLSE32 bestimmen. Starten Sie dazu den PDnetIP-Controller im Lade-Modus und führen Sie die Funktion „`Online`“ / „`Info`“ aus.

Mit DLSE32 kann nur die Lader-Hauptversion bestimmt werden. Wurde die Version x.y geladen, erkennt DLSE32 Version x.

5.2.3 NetPro Projektierung

Die Projektierung des PAD-800e erfolgt mit Ausnahme der PMB-Adresse mit NetPro. Die dafür notwendigen Informationen zur Projektierung können dem Anwenderhandbuch von NetPro entnommen werden.

Wichtig! Frühere Anlagen wurden mit DLSE projektiert. Dies gilt nicht für den aktuell gelieferten Firmware-Stand. Beachten Sie bei Altanlagen die Hinweise in Kapitel 5.2.5.

5.2.4 ICOS Projektierung

Die Projektierung der ICOS spezifischen Informationen erfolgt mit COM800. Die Projektierungsinformationen werden bei einem IK von der ALU an den PDnetIP-Controller übergeben. Dabei sind für den PAD-800e nur wenige Angaben der ICOS Projektierung für den Betrieb notwendig. Die folgende Tabelle gibt einen Überblick darüber, welche Kommunikationsparameter aus der Projektierung vom PAD-800e ausgewertet werden:

- **Prozedur**
  Die ICOS Prozedur-Kennung entscheidet darüber wie der PDnetIP-Controller einige Informationen interpretiert. Je nach verwendeter Prozedur-Kennung wird als BroadCast-Adresse 127 oder 255 interpretiert.
- **Adress-Byte**
  Es gibt eine Sonderbehandlung des A-Byte, wenn das Protokoll kein A-Byte für die verwendete Prozedur vorsieht. In diesem Fall wird nach einem Regelsatz das A-Byte der Station bestimmt. Wird eine Slave-Prozedur wie BUS_1_SL oder STERN_1_SL projektiert, benutzt der PDnetIP-Controller immer das A-Byte 1. Arbeitet der PDnetIP-Controller mit einer Master-Prozedur (BUS_1_MA, STERN_1_MA), benutzt er immer das A-Byte 240.
- **Sendepuffer+Empfangspuffer**
  Die bei den Original-Baugruppen bestehenden Konventionen zur Größe des Sende- und Empfangspuffers gelten nicht für die PDnetIP-Controller.

5.2.5 Aktualisierung vorhandener Systeme

Vorhandene Installationen eines PDnetIP-Controllers PAD-800e können auf den aktuellen Lieferstand aufgerüstet werden. Dazu sind folgende Punkte zu prüfen:

- Ist der Lader aktuell? Die aktuelle Firmware benötigt einen aktuellen Lader, um die Setupdaten programmieren zu können. Siehe Kapitel 7.3.
- Läuft die Anlage mit einer Firmware ab Version 4.xx, genügt es die Setupdaten mit NetPro neu zu programmieren sowie die Firmware zu aktualisieren. Wurde die Projektierung noch mit DLSE durchgeführt, muss mit NetPro ein neues Projekt angelegt werden.

Achtung! In jedem Fall ist es notwendig, den alten Anlagenzustand so zu sichern, dass dieser wieder aktiviert werden kann. Sind die dafür notwendigen Informationen, Softwarestände und Werkzeuge nicht vorhanden, ist eine Umrüstung nicht sinnvoll.

Achtung! Die aktuelle Firmware bezieht sämtliche Konfigurationsinformationen aus den mit NetPro programmierten Setupdaten. Das Verfahren der Projektierung via DLSE wird nicht mehr verwendet und von der aktuellen Firmware nicht unterstützt. So werden die Menüpunkte zur Konfiguration der IP-Adresse bzw. des Endgeräteinterfaces in DLSE auch nicht mehr angeboten.

5.3 Strukturen

Die Kommunikation von Strukturen steht für den PAD-800e derzeit nicht zur Verfügung.

6 Beispiele

6.1 ModBus/TCP

6.1.1 ModBus-Master Auftragsblock

Nachfolgend werden einige Beispiele für die Anwendung der Master-Funktion gegeben. Die Beispiele beschreiben die Funktionalität und wie der Auftragsblock dazu belegt werden muss. Die im Anwenderprogramm der SPS notwendige Ablaufsteuerung im Statuswort wird nicht beschrieben. Für Testzwecke können die Auftragsblöcke auch über die OnLine-Liste (AKF35) gestartet werden.

Achtung! Die im Auftragsblock gemachten Adressangaben (Wort 4) beziehen sich auf ein Mapping und geben keine feste lokale Adresse an.

6.1.1.1 Read Register

Im Beispiel werden aus der ModBus/TCP Station mit der IP-Adresse 192.168.1.33 aus dem Register 500, 10 Register gelesen und in der lokalen SPS ab Register 2000 abgelegt.

```
W1     Statuswort          2
W2     Funktionscode       1
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      2000
W6     Anzahl              10
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   500
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```

6.1.1.2 Write Register

Im Beispiel wird ein Schreibbefehl in die ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ausgelöst. Es werden 25 Merkerworte ab Register 100 in die andere ModBus/TCP-Station ab Register 500 geschrieben.

```
W1     Statuswort          2
W2     Funktionscode       2
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      100
W6     Anzahl              25
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   500
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```

6.1.1.3 Read Coil

Im Beispiel werden aus der ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ab dem Coil 800, 10 Coils gelesen und in der lokalen SPS ab Coil 55 abgelegt.

```
W1     Statuswort          2
W2     Funktionscode       3
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      55
W6     Anzahl              10
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   800
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```

6.1.1.4 Write Coil

Im Beispiel wird ein Schreibbefehl in die ModBus/TCP Station mit der IP-Adresse 192.168.1.33 ausgelöst. Es werden 30 Coils ab Coil 500 in die andere ModBus/TCP-Station ab Coil 310 geschrieben.

```
W1     Statuswort          2
W2     Funktionscode       4
W3     Datentyp            0
W4     ErwAdresse          0
W5     Lokale Adresse      500
W6     Anzahl              30
W7     IP-Adresse 1        192
W8     IP-Adresse 2        168
W9     IP-Adresse 3        1
W10    IP-Adresse 4        33
W11    UnitID              0
W12    TRID                0
W13    Entfernte Adresse   310
W14    TimeOut             0
W15    Fehlerwort          0
W16    Info                0
```

7 Aktualisierung

Alle nachfolgend beschriebenen Aktualisierungen benötigen den PAD-800e im Lade-Modus. Wie der Lade-Modus im PDnetIP-Controller aktiviert wird, ist in Kapitel 3.3.2.1 beschrieben.

Im Flashmemory des PDnetIP-Controllers werden in separaten Bereichen die Firmware, die Projektierung sowie der Lader abgelegt. Werden in einem PDnetIP-Controller neue Setupdaten geladen, sind die anderen Bereiche (Lader, Firmware) davon nicht betroffen.

7.1 Projektierung

Die Projektierung der PDnetIP-Controller erfolgt mit NetPro. Jede in NetPro erfolgte Konfigurationsänderung muss über die serielle Schnittstelle in den PDnetIP-Controller übertragen werden.

Um die Projektierung einer Station zu programmieren, sind folgende Schritte in NetPro notwendig:

- Menüpunkt „`Projekt`“ / „`Öffnen`“ auswählen.
- Menüpunkt „`Bearbeiten`“ / „`Stationen`“ auswählen.
- Mit der rechten Maustaste auf die zu programmierende Station klicken.
- Den Menüpunkt „`Programmieren über..`“ / „`serielle Schnittstelle`“ auswählen.
- Anschließend die serielle Schnittstelle des PCs, mit welcher der PAD-800e verbunden ist, auswählen. Prüfen Sie die ausgewählte Geschwindigkeit, welche von der Lader-Version abhängig ist (Kapitel 7.3.1).

7.2 Firmware

Die Aktualisierung der Firmware erfolgt mit DLSE32. Beachten Sie, dass bei einem Firmwarewechsel ggf. auch die Setupdaten mit einer passenden NetPro-Version neu programmiert werden müssen.

Um eine neue Firmware zu laden, sind folgende Schritte in DLSE32 notwendig:

- Menüpunkt „`Firmware`" / „`Auswählen`" auswählen.
- Download der Firmware über den Menüpunkt „`Online`" / „`Download`" auswählen.
- Nach einem erfolgreichen Download führt der PDnetIP-Controller einen Neustart durch.

7.3 Lader

Welche Lader-Version eingesetzt wird, können Sie mit DLSE32 bestimmen. Starten Sie dazu den PDnetIP-Controller im Lade-Modus und führen Sie die Funktion „`Online`“ / „`Info`“ aus.

Mit DLSE32 kann nur die Lader-Hauptversion bestimmt werden. Wurde die Version x.y geladen, erkennt DLSE32 Version x.

Wird eine ältere Lader-Version eingesetzt, kann diese mit einer Spezial-Firmware aktualisiert werden. Diese befindet sich auf der PDnetIP-CD in dem Verzeichnis „`\Software\Lader Update`“. Der aktuelle Lader kann zusammen mit älteren Firmware-Versionen eingesetzt werden.

Achtung! Der Einsatz dieser Firmware erfolgt auf eigenes Risiko. Eine Anwendung ohne Ersatzkarte ist in jedem Fall zu vermeiden. Scheitert die Aktualisierung ist der PDnetIP-Controller nicht mehr funktionsfähig! Zur Reaktivierung muss der PDnetIP-Controller eingesandt werden.

Wurde mit DLSE32 die Spezial-Firmware zum Lader-UpDate geladen, erfolgt die Aktualisierung nicht automatisch. Drücken Sie dazu den Taster solange, bis der Lader erneut aktiviert wurde.

Achtung! Bei einem Lader-UpDate wird die Firmware im PDnetIP-Controller überschrieben. Stellen Sie die Verfügbarkeit der zuvor geladenen Firmware sicher, um diese nach dem Lader-UpDate erneut laden zu können.

7.3.1 Lader-Versionen

Die folgende Tabelle gibt eine Übersicht über die Baudrate, mit welcher der Lader über die serielle Schnittstelle kommuniziert. Die genutzte Baudrate muss bei der Konfiguration von DLSE32 bzw. NetPro beachtet werden.

<table>
<tr><th>Lader-Version</th><th>Seriell 1</th><th>Seriell 2</th></tr>
<tr><td>1</td><td rowspan="2">19.200 Baud</td><td>19.200 Baud</td></tr>
<tr><td>2..5</td><td rowspan="2">115.200 Baud</td></tr>
<tr><td>6..8</td><td>115.200 Baud</td></tr>
</table>

8 Anhang

8.1 Fehlerliste TCP/IP-Stack

Die hier aufgeführten Fehlermeldungen sind bei der Diagnose hilfreich. Einige Firmware-Module (ICOS-Diagnose, Ausgaben der Diagnose-Firmware, SYSLOG-Meldungen) protokollieren unter Umständen Fehlercodes des TCP/IP-Stack.

Fehler	Beschreibung
101	endpoint address not available
102	address in use
103	family not supported
104	arp table full
105	invalid baud rate
106	invalid comm port number
107	invalid device type
108	invalid interface number
109	invalid mask (ether must not be all f´s)
110	invalid ping response
111	endpoint refused connection
112	destination address is required
113	destination unreachable (ICMP)
114	invalid parameter (pointer is 0)
115	interface closed
116	interface table full
117	interface open failed
118	operation (connect) is in progress
119	invalid function call (parameter)

8 Anhang
8.1 Fehlerliste TCP/IP-Stack

Fehler	Beschreibung
120	socket is already connected
121	multicast table full
122	multicast address not found
123	out of ports
124	network is down (send failed)
125	network unreachable (keepalive failed)
126	out of DCUs (packets)
127	option parameter is invalid
128	socket is not connected
129	RTIP not initialized
130	invalid socket descriptor
131	not enough devices
132	socket type or specified operation not
133	send failed due to output list
134	could not determine device
135	non-reentrancy error
136	routing table entry not found
137	routing table full
138	resource initialization failed
139	illegal operation due to socket shutdown
140	Timeout
141	type not supported
142	send needs to ARP but ARP is disabled
143	not enough heap memory for allocate request
144	table full (such as ARP cache, device table etc)
145	Illegal sized packet
146	device open failed
200	The global host table is full
201	The host table entry wasn't found
202	To many dns-servers
203	DNS server returned error
204	DNS name is too long
205	DNS call to socket function failed
206	data not available
207	socket call within DNS failed; probably no response received from server
208	No server in server table

8.2 Konfiguration PMB-Adresse

Die Konfiguration der Adresslage am PMB-Bus ist abhängig von der Hardware-Revision des PAD-800e. Bestimmen Sie zuerst die Hardware-Revision und wählen anschließend die gewünschte Konfiguration aus.

In den folgenden Tabellen wird für die Position „ON“ eines DIP-Schalters die Zahl 1 und für die Position OFF die Zahl 0 verwendet.

Achtung! Der PAD-800e darf nur gezogen/gesteckt werden, wenn die Steuerung spannungslos ist.

8.2.1 Hardware-Revision 1, 2

Die Hardware-Revisionen 1 und 2 unterstützten die Adressierung eines 8k Bereiches auf dem PMB.

Der Speicherbereich von 8 KByte wird über die DIP-Schalter 1..6 konfiguriert. Der DIP-Schalter 8 bestimmt, wie viele Adressleitungen (A0..A15 oder A0..A18) vom PAD-800e ausgewertet werden. Dies ist für Zentralbaugruppen von Bedeutungen, welche nicht den vollen Adressraum unterstützen.

DIP-Schalter 1..6 – PMB Adresskonfiguration

DIP-Schalter	Adressleitung	Adresse
1	A18	0x40000
2	A17	0x20000
3	A16	0x10000
4	A15	0x08000
5	A14	0x04000
6	A13	0x02000

Die Adresse ist aktiviert wenn der DIP-Schalter auf **ON** steht.

DIP Schalter 7 – reserviert

DIP-Schalter	Position	Beschreibung
7	**OFF**	Feste Einstellung

DIP Schalter 8 – PMB Adressraum

DIP-Schalter	Position	Beschreibung
8	ON	Auswertung der Adressleitungen A0..A15. Diese Einstellung wird für ältere Zentralbaugruppe wie der ALU 812 benötigt.
	OFF	Auswertung Adressleitungen A0..A18 Diese Einstellung wird für die Nachfolgemodelle wie der ALU 981 benötigt.

8.2.1.1 Einstellungen für 8k Speicherbereich

Der DIP-Schalter 8 ist in dieser Tabelle nicht enthalten, da dieser je nach gewählter Zentralbaugruppe zu setzen ist.

1	2	3	4	5	6	7	Adresslage
0	0	0	0	0	0	0	0x00000
0	0	0	0	0	1	0	0x02000
0	0	0	0	1	0	0	0x04000
0	0	0	0	1	1	0	0x06000
0	0	0	1	0	0	0	0x08000
0	0	0	1	0	1	0	0x0A000
0	0	0	1	1	0	0	0x0C000
0	0	0	1	1	1	0	0x0E000
0	0	1	0	0	0	0	0x10000
0	0	1	0	0	1	0	0x12000
0	0	1	0	1	0	0	0x14000
0	0	1	0	1	1	0	0x16000
0	0	1	1	0	0	0	0x18000
0	0	1	1	0	1	0	0x1A000
0	0	1	1	1	0	0	0x1C000
0	0	1	1	1	1	0	0x1E000
0	1	0	0	0	0	0	0x20000
0	1	0	0	0	1	0	0x22000
0	1	0	0	1	0	0	0x24000
0	1	0	0	1	1	0	0x26000
0	1	0	1	0	0	0	0x28000
0	1	0	1	0	1	0	0x2A000
0	1	0	1	1	0	0	0x2C000
0	1	0	1	1	1	0	0x2E000
0	1	1	0	0	0	0	0x30000
0	1	1	0	0	1	0	0x32000
0	1	1	0	1	0	0	0x34000
0	1	1	0	1	1	0	0x36000
0	1	1	1	0	0	0	0x38000
0	1	1	1	0	1	0	0x3A000
0	1	1	1	1	0	0	0x3C000
0	1	1	1	1	1	0	0x3E000
1	0	0	0	0	0	0	0x40000
1	0	0	0	0	1	0	0x42000
1	0	0	0	1	0	0	0x44000
1	0	0	0	1	1	0	0x46000
1	0	0	1	0	0	0	0x48000
1	0	0	1	0	1	0	0x4A000
1	0	0	1	1	0	0	0x4C000
1	0	0	1	1	1	0	0x4E000
1	0	1	0	0	0	0	0x50000
1	0	1	0	0	1	0	0x52000
1	0	1	0	1	0	0	0x54000
1	0	1	0	1	1	0	0x56000
1	0	1	1	0	0	0	0x58000
1	0	1	1	0	1	0	0x5A000
1	0	1	1	1	0	0	0x5C000
1	0	1	1	1	1	0	0x5E000

1	2	3	4	5	6	7	Adresslage
1	1	0	0	0	0	0	0x60000
1	1	0	0	0	1	0	0x62000
1	1	0	0	1	0	0	0x64000
1	1	0	0	1	1	0	0x66000
1	1	0	1	0	0	0	0x68000
1	1	0	1	0	1	0	0x6A000
1	1	0	1	1	0	0	0x6C000
1	1	0	1	1	1	0	0x6E000
1	1	1	0	0	0	0	0x70000
1	1	1	0	0	1	0	0x72000
1	1	1	0	1	0	0	0x74000
1	1	1	0	1	1	0	0x76000
1	1	1	1	0	0	0	0x78000
1	1	1	1	0	1	0	0x7A000
1	1	1	1	1	0	0	0x7C000
1	1	1	1	1	1	0	0x7E000

8.2.2 Hardware-Revision 3, 4, 5

Der Basisadresse des DPM im Adressraum der Steuerung wird über die DIP-Schalter 2..7 konfiguriert. Zusätzlich kann die Größe des Interface statt 8k auch für 4k konfiguriert werden. In diesem Fall kann die Adresse zusätzlich noch um 4k verschoben werden, um eine beliebige Staffelung der Adressen in 4k-Schritten zu ermöglichen. Die Auswahl des unterstützten Adressraums von A0..A15 bzw. A0..A18 ist ab Hardware-Revision 3 nicht mehr notwendig.

DIP-Schalter 2..7 – PMB Adresskonfiguration

DIP-Schalter	Adressleitung	Adresse
7	A18	0x40000
6	A17	0x20000
5	A16	0x10000
4	A15	0x08000
3	A14	0x04000
2	A13	0x02000

Die Adresse ist aktiviert wenn der DIP-Schalter auf **OFF** steht.

DIP-Schalter 1 und 8

DSW1	DSW8	Beschreibung
ON	ON	4k-Interface, A12=0
OFF		4k-Interface, A12=1
ON	**OFF**	8k-Interface (Werkskonfiguration)
OFF		Verboten

DIP-Schalter 9 und 10

DIP-Schalter	Position	Beschreibung
9	**ON**	Feste Einstellung
10	**OFF**	Feste Einstellung

8.2.2.1 Einstellungen für 4k Speicherbereich

Die Verwendung eines 4k-Speicherbereiches ist ab Hardware-Revision 4 möglich.

1	2	3	4	5	6	7	8	9	10	Adresslage
1	1	1	1	1	1	1	1	1	0	0x00000
0	1	1	1	1	1	1	1	1	0	0x01000
1	0	1	1	1	1	1	1	1	0	0x02000
0	0	1	1	1	1	1	1	1	0	0x03000
1	1	0	1	1	1	1	1	1	0	0x04000
0	1	0	1	1	1	1	1	1	0	0x05000
1	0	0	1	1	1	1	1	1	0	0x06000
0	0	0	1	1	1	1	1	1	0	0x07000
1	1	1	0	1	1	1	1	1	0	0x08000
0	1	1	0	1	1	1	1	1	0	0x09000
1	0	1	0	1	1	1	1	1	0	0x0A000
0	0	1	0	1	1	1	1	1	0	0x0B000
1	1	0	0	1	1	1	1	1	0	0x0C000
0	1	0	0	1	1	1	1	1	0	0x0D000
1	0	0	0	1	1	1	1	1	0	0x0E000
0	0	0	0	1	1	1	1	1	0	0x0F000
1	1	1	1	0	1	1	1	1	0	0x10000
0	1	1	1	0	1	1	1	1	0	0x11000
1	0	1	1	0	1	1	1	1	0	0x12000
0	0	1	1	0	1	1	1	1	0	0x13000
1	1	0	1	0	1	1	1	1	0	0x14000
0	1	0	1	0	1	1	1	1	0	0x15000
1	0	0	1	0	1	1	1	1	0	0x16000
0	0	0	1	0	1	1	1	1	0	0x17000
1	1	1	0	0	1	1	1	1	0	0x18000
0	1	1	0	0	1	1	1	1	0	0x19000
1	0	1	0	0	1	1	1	1	0	0x1A000
0	0	1	0	0	1	1	1	1	0	0x1B000
1	1	0	0	0	1	1	1	1	0	0x1C000
0	1	0	0	0	1	1	1	1	0	0x1D000
1	0	0	0	0	1	1	1	1	0	0x1E000
0	0	0	0	0	1	1	1	1	0	0x1F000
1	1	1	1	1	0	1	1	1	0	0x20000
0	1	1	1	1	0	1	1	1	0	0x21000
1	0	1	1	1	0	1	1	1	0	0x22000
0	0	1	1	1	0	1	1	1	0	0x23000
1	1	0	1	1	0	1	1	1	0	0x24000
0	1	0	1	1	0	1	1	1	0	0x25000
1	0	0	1	1	0	1	1	1	0	0x26000
0	0	0	1	1	0	1	1	1	0	0x27000
1	1	1	0	1	0	1	1	1	0	0x28000
0	1	1	0	1	0	1	1	1	0	0x29000
1	0	1	0	1	0	1	1	1	0	0x2A000
0	0	1	0	1	0	1	1	1	0	0x2B000
1	1	0	0	1	0	1	1	1	0	0x2C000
0	1	0	0	1	0	1	1	1	0	0x2D000
1	0	0	0	1	0	1	1	1	0	0x2E000
0	0	0	0	1	0	1	1	1	0	0x2F000

1	2	3	4	5	6	7	8	9	10	Adresslage
1	1	1	1	0	0	1	1	1	0	0x30000
0	1	1	1	0	0	1	1	1	0	0x31000
1	0	1	1	0	0	1	1	1	0	0x32000
0	0	1	1	0	0	1	1	1	0	0x33000
1	1	0	1	0	0	1	1	1	0	0x34000
0	1	0	1	0	0	1	1	1	0	0x35000
1	0	0	1	0	0	1	1	1	0	0x36000
0	0	0	1	0	0	1	1	1	0	0x37000
1	1	1	0	0	0	1	1	1	0	0x38000
0	1	1	0	0	0	1	1	1	0	0x39000
1	0	1	0	0	0	1	1	1	0	0x3A000
0	0	1	0	0	0	1	1	1	0	0x3B000
1	1	0	0	0	0	1	1	1	0	0x3C000
0	1	0	0	0	0	1	1	1	0	0x3D000
1	0	0	0	0	0	1	1	1	0	0x3E000
0	0	0	0	0	0	1	1	1	0	0x3F000
1	1	1	1	1	1	0	1	1	0	0x40000
0	1	1	1	1	1	0	1	1	0	0x41000
1	0	1	1	1	1	0	1	1	0	0x42000
0	0	1	1	1	1	0	1	1	0	0x43000
1	1	0	1	1	1	0	1	1	0	0x44000
0	1	0	1	1	1	0	1	1	0	0x45000
1	0	0	1	1	1	0	1	1	0	0x46000
0	0	0	1	1	1	0	1	1	0	0x47000
1	1	1	0	1	1	0	1	1	0	0x48000
0	1	1	0	1	1	0	1	1	0	0x49000
1	0	1	0	1	1	0	1	1	0	0x4A000
0	0	1	0	1	1	0	1	1	0	0x4B000
1	1	0	0	1	1	0	1	1	0	0x4C000
0	1	0	0	1	1	0	1	1	0	0x4D000
1	0	0	0	1	1	0	1	1	0	0x4E000
0	0	0	0	1	1	0	1	1	0	0x4F000
1	1	1	1	0	1	0	1	1	0	0x50000
0	1	1	1	0	1	0	1	1	0	0x51000
1	0	1	1	0	1	0	1	1	0	0x52000
0	0	1	1	0	1	0	1	1	0	0x53000
1	1	0	1	0	1	0	1	1	0	0x54000
0	1	0	1	0	1	0	1	1	0	0x55000
1	0	0	1	0	1	0	1	1	0	0x56000
0	0	0	1	0	1	0	1	1	0	0x57000
1	1	1	0	0	1	0	1	1	0	0x58000
0	1	1	0	0	1	0	1	1	0	0x59000
1	0	1	0	0	1	0	1	1	0	0x5A000
0	0	1	0	0	1	0	1	1	0	0x5B000
1	1	0	0	0	1	0	1	1	0	0x5C000
0	1	0	0	0	1	0	1	1	0	0x5D000
1	0	0	0	0	1	0	1	1	0	0x5E000
0	0	0	0	0	1	0	1	1	0	0x5F000
1	1	1	1	1	0	0	1	1	0	0x60000
0	1	1	1	1	0	0	1	1	0	0x61000
1	0	1	1	1	0	0	1	1	0	0x62000
0	0	1	1	1	0	0	1	1	0	0x63000
1	1	0	1	1	0	0	1	1	0	0x64000
0	1	0	1	1	0	0	1	1	0	0x65000

1	2	3	4	5	6	7	8	9	10	Adresslage
1	0	0	1	1	0	0	1	1	0	0x66000
0	0	0	1	1	0	0	1	1	0	0x67000
1	1	1	0	1	0	0	1	1	0	0x68000
0	1	1	0	1	0	0	1	1	0	0x69000
1	0	1	0	1	0	0	1	1	0	0x6A000
0	0	1	0	1	0	0	1	1	0	0x6B000
1	1	0	0	1	0	0	1	1	0	0x6C000
0	1	0	0	1	0	0	1	1	0	0x6D000
1	0	0	0	1	0	0	1	1	0	0x6E000
0	0	0	0	1	0	0	1	1	0	0x6F000
1	1	1	1	0	0	0	1	1	0	0x70000
0	1	1	1	0	0	0	1	1	0	0x71000
1	0	1	1	0	0	0	1	1	0	0x72000
0	0	1	1	0	0	0	1	1	0	0x73000
1	1	0	1	0	0	0	1	1	0	0x74000
0	1	0	1	0	0	0	1	1	0	0x75000
1	0	0	1	0	0	0	1	1	0	0x76000
0	0	0	1	0	0	0	1	1	0	0x77000
1	1	1	0	0	0	0	1	1	0	0x78000
0	1	1	0	0	0	0	1	1	0	0x79000
1	0	1	0	0	0	0	1	1	0	0x7A000
0	0	1	0	0	0	0	1	1	0	0x7B000
1	1	0	0	0	0	0	1	1	0	0x7C000
0	1	0	0	0	0	0	1	1	0	0x7D000
1	0	0	0	0	0	0	1	1	0	0x7E000

8.2.2.2 Einstellungen für 8k Speicherbereich

1	2	3	4	5	6	7	8	9	10	Adresslage
1	1	1	1	1	1	1	0	1	0	0x00000
1	0	1	1	1	1	1	0	1	0	0x02000
1	1	0	1	1	1	1	0	1	0	0x04000
1	0	0	1	1	1	1	0	1	0	0x06000
1	1	1	0	1	1	1	0	1	0	0x08000
1	0	1	0	1	1	1	0	1	0	0x0A000
1	1	0	0	1	1	1	0	1	0	0x0C000
1	0	0	0	1	1	1	0	1	0	0x0E000
1	1	1	1	0	1	1	0	1	0	0x10000
1	0	1	1	0	1	1	0	1	0	0x12000
1	1	0	1	0	1	1	0	1	0	0x14000
1	0	0	1	0	1	1	0	1	0	0x16000
1	1	1	0	0	1	1	0	1	0	0x18000
1	0	1	0	0	1	1	0	1	0	0x1A000
1	1	0	0	0	1	1	0	1	0	0x1C000
1	0	0	0	0	1	1	0	1	0	0x1E000
1	1	1	1	1	0	1	0	1	0	0x20000
1	0	1	1	1	0	1	0	1	0	0x22000
1	1	0	1	1	0	1	0	1	0	0x24000
1	0	0	1	1	0	1	0	1	0	0x26000
1	1	1	0	1	0	1	0	1	0	0x28000
1	0	1	0	1	0	1	0	1	0	0x2A000
1	1	0	0	1	0	1	0	1	0	0x2C000
1	0	0	0	1	0	1	0	1	0	0x2E000

8 Anhang
8.2 Konfiguration PMB-Adresse

1	2	3	4	5	6	7	8	9	10	Adresslage
1	1	1	1	0	0	1	0	1	0	0x30000
1	0	1	1	0	0	1	0	1	0	0x32000
1	1	0	1	0	0	1	0	1	0	0x34000
1	0	0	1	0	0	1	0	1	0	0x36000
1	1	1	0	0	0	1	0	1	0	0x38000
1	0	1	0	0	0	1	0	1	0	0x3A000
1	1	0	0	0	0	1	0	1	0	0x3C000
1	0	0	0	0	0	1	0	1	0	0x3E000
1	1	1	1	1	1	0	0	1	0	0x40000
1	0	1	1	1	1	0	0	1	0	0x42000
1	1	0	1	1	1	0	0	1	0	0x44000
1	0	0	1	1	1	0	0	1	0	0x46000
1	1	1	0	1	1	0	0	1	0	0x48000
1	0	1	0	1	1	0	0	1	0	0x4A000
1	1	0	0	1	1	0	0	1	0	0x4C000
1	0	0	0	1	1	0	0	1	0	0x4E000
1	1	1	1	0	1	0	0	1	0	0x50000
1	0	1	1	0	1	0	0	1	0	0x52000
1	1	0	1	0	1	0	0	1	0	0x54000
1	0	0	1	0	1	0	0	1	0	0x56000
1	1	1	0	0	1	0	0	1	0	0x58000
1	0	1	0	0	1	0	0	1	0	0x5A000
1	1	0	0	0	1	0	0	1	0	0x5C000
1	0	0	0	0	1	0	0	1	0	0x5E000
1	1	1	1	1	0	0	0	1	0	0x60000
1	0	1	1	1	0	0	0	1	0	0x62000
1	1	0	1	1	0	0	0	1	0	0x64000
1	0	0	1	1	0	0	0	1	0	0x66000
1	1	1	0	1	0	0	0	1	0	0x68000
1	0	1	0	1	0	0	0	1	0	0x6A000
1	1	0	0	1	0	0	0	1	0	0x6C000
1	0	0	0	1	0	0	0	1	0	0x6E000
1	1	1	1	0	0	0	0	1	0	0x70000
1	0	1	1	0	0	0	0	1	0	0x72000
1	1	0	1	0	0	0	0	1	0	0x74000
1	0	0	1	0	0	0	0	1	0	0x76000
1	1	1	0	0	0	0	0	1	0	0x78000
1	0	1	0	0	0	0	0	1	0	0x7A000
1	1	0	0	0	0	0	0	1	0	0x7C000
1	0	0	0	0	0	0	0	1	0	0x7E000